introduzione alla programmazione

OLIVETTI
GENERAL ELECTRIC

GREGORIO — GIANCARLO

I N D I C E

PARTE I^ N O Z I O N I I N T R O D U T T I V E

CAP. 1°

STORIA E SVILUPPO DEI CALCOLATORI pag. 1

1.1 Matematica e calcolo " 1
1.2 Strumenti primitivi di calcolo " 1
1.3 Le prime calcolatrici meccaniche " 2
1.4 La produzione industriale dei calcolatori e degli elaboratori " 4
1.5 Il relay e i componenti elettronici " 5
1.6 Gli eleboratori elettronici " 6
1.7 Conclusioni " 7

CAP. 2°

SISTEMI DI NUMERAZIONE pag. 11

2.1 Il sistema decimale " 11
2.2 Il sistema binario " 13
2.3 Sistema decimale o sistema binario " 16

CAP. 3°

ELEMENTI DELL'ALGEBRA DI BOOLE pag. 19

3.1 Generalità " 19
3.2 Addizione o Riunione (simbolo + oppure U) " 19
3.3 Moltiplicazione o intersezione (simbolo x o ∩) " 21
3.4 Inversione d'una somma o di un prodotto " 23

CAP. 4°

	IL LINGUAGGIO DEGLI ELABORATORI	pag.	25
4.1	I linguaggi in generale	"	25
4.2	Il bit	"	26
4.3	Il bit e i caratteri	"	27
4.4	Scelta delle configurazioni	"	29
4.5	Registrazione e lettura dei caratteri	"	29
4.6	Serie e parallelo	"	30
4.7	Il bit di controllo o chech	"	30

PARTE II° L'E L A B O R A T O R E E I S U O I O R G A N I

CAP. 1°

	SISTEMA INTEGRATO	pag.	33
1.1	Distinzione fra calcolatore e elaboratore	"	33
1.2	Schema generale di un elaboratore	"	33
1.3	Le Unità di Entrata e di Uscita	"	34
1.4	L'Unità Centrale	"	36

CAP. 2°

	I SUPPORTI	pag.	38
2.1	Definizione	"	38
2.2	Schede perforate	"	48

2.3 I codici scheda pag. 39
2.4 Perforazione della scheda " 40
2.5 Lettura delle schede " 41
2.6 La banda perforata " 42
2.7 Registrazione su banda " 43
2.8 Lettura su banda " 43
2.9 Il nastro magnetico " 44
2.10 L'introduzione magnetica e il ciclo di isteresi " 44
2.11 Le correnti indotte " 46
2.12 Registrazione e lettura su nastro magnetico " 46
2.13 Il bit di temporizzazione " 47
2.14 Organizzazione dei dati su nastro " 48
2.15 Gli interblocchi " 49
2.16 Caratteri particolari di nastro " 50
2.17 Il modulo stampa " 51
2.18 Alcune considerazioni pratiche " 52

CAP. 3°

LE MEMORIE pag. 55

3.1 Generalità " 55
3.2 Memorie ad accesso sequenziale ed ad accesso casuale " 56
3.3 L'indirizzo " 57
3.4 Individuazione di una parola in memoria " 57
3.5 Il tempo di accesso " 59
3.6 La circolarità delle memorie " 59
3.7 Modularità delle memorie " 60
3.8 Classificazione dei vari tipi di memoria " 61

3.9 Memorie a tamburo magnetico pag. 62
3.10 Memorie a dischi magnetici " 63
3.11 Memorie a nuclei magnetici " 64
3.12 Registrazione di un carattere nelle memorie a nuclei " 64
3.13 Lettura di un carattere nelle memorie a nuclei " 65
3.14 L'indirizzamento nelle memorie a nuclei " 66

CAP. 4°

GLI ALTRI ORGANI DELL'UNITA' CENTRALE pag. 69

4.1 Unità aritmetica e logica " 69
4.2 L' accomulatore " 71
4.3 Indirizzatore di programma " 71
4.4 Indirizzatore di operando " 72
4.5 Organo di temporizzazione " 73
4.6 Unità di Governo " 73
4.7 I canali di collegamento " 74

CAP. 5°

LE ISTRUZIONI pag. 76

5.1 Definizione " 76
5.2 Struttura di una istruzione " 76
5.3 Registrazione delle istruzioni " 76
5.4 Componenti dell'Istruzione " 77
5.5 Indirizzo dell' Istruzione " 78
5.6 Fase preparatoria e fase esecutiva

CAP. 6°

SVOLGIMENTO DI ISTRUZIONI IN UN ELABORATORE IDEALE pag. 81

PARTE III^ L A P R O G R A M M A Z I O N E

CAP. 1°

PROGRAMMAZIONE pag. 97

1.1 Funzione generale di un elaboratore " 97
1.2 Le funzioni logiche e operative di un elaboratore sono elementari " 99
1.3 Necessità di precisione " 100
1.4 Convenienza del problema da risolvere la sua natura ciclica " 101
1.5 Definizione di programma " 102
1.6 Le tre fasi di risoluzione di un problema " 103

CAP. 2°

L'ANALISI pag. 106

2.1 Fase di analisi " 106
2.2 Definizione dei supporti e delle fasi di lavoro " 108
2.3 Diagramma e blocchi per il flusso dei dati " 109
2.4 Simboli di dati " 110

2.5 Simboli di operazione pag. 110
2.6 Linee di flusso " 111
2.7 Simboli ausiliari " 111

CAP. 3°

LINEA OPERATIVA DI PROGRAMMA: IL DIAGRAMMA DI FLUSSO pag. 113
3.1 Necessità di una linea operativa " 113
3.2 L'importanza dell' enunciato d'analisi " 113
3.3 Diagrammi a blocchi per il programma " 114
3.4 Simboli di operazione " 115
3.5 Simboli relativi al flusso delle operazioni " 116
3.6 Simboli ausiliari " 116
3.7 Norme per la corretta applicazione di alcuni simboli " 117
3.8 Esempio di diagramma di flusso " 121
3.9 I cicli iterativi " 122
3.10 Prova logica dei diagrammi di flusso " 124
3.11 Esempio di prova logica " 125
3.12 Ottimizzazione di un diagramma di flusso " 129
3.13 Differenti livelli di diagrammi di flusso " 130

CAP. 4°

I SOTTOPROGRAMMI O SUBROUTINES pag. 133
4.1 Introduzione e definizione " 133

4.2 Caratteristiche dei sottoprogrammi pag. 134
4.3 Biblioteche di sottoprogrammi " 134

CAP. 5°

LINEA OPERATIVA DI MACCHINA E CODIFICAZIONE pag. 136

5.1 Introduzione " 136
5.2 Il linguaggio base " 136
5.3 I linguaggi simbolici " 137
5.4 Linguaggi simbolici da un punto di vista logico " 138
5.5 I linguaggi simbolici dal punto di vista dei programmi traduttori

P A R T E P R I M A

NOZIONI INTRODUTTIVE

I STORIA E SVILUPPO DEI CALCOLATORI

1.1 Matematica e calcolo

Strettissimi sono i legami che esistono fra matematica e calcolo, specialmente nel campo delle scienze applicate.

Fin dai lontani tempi degli Egizi, degli Assiri, dei Greci e dei Romani, la risoluzione dei problemi di matematica e di ingegneria comportava l'attenta e controllata esecuzione di un rilevante numero di calcoli. Al fascino e alla bellezza di tali problemi si opponeva il peso e la noia del computo delle operazioni.

Di lì la necessità di escogitare strumenti, sia pure semplici e rudimentali, che valessero a rendere meno arduo il cammino che separava dalla meta.

1.2 Strumenti primitivi di calcolo

L'uomo cominciò a rappresentare i numeri colle dita o con oggetti embrionali equivalenti: sassolini, chicchi di grano, cannucce, piccoli giunchi. L'Abaco a pallottoliere è il primo rudimentale strumento, formato da palline scorrevoli dentro un filo teso. Diviso in due parti, esso consentiva di esprimere i vari numeri nel modo indicato dalla figura 1. La pallina isolata, a seconda che sia spostata in un senso o nell'altro, indica numeri minori di 5 o superiori a 4. Le altre quattro palline, col loro spostamento, esprimono quante unità occorre sommare al valore indicato dalla pallina isolata. Il totale dà il numero desiderato.

Gli strumenti finora esaminati sono evidentemente molto lontani dalla concezione delle moderne macchine calcolatrici.

Il sassolino, la pallina o il chicco di grano per spostarsi richiedono l'intervento dell'uomo e tanto la loro velocità, quanto la loro esattezza dipendono completamente dall'operatore.

1.3 Le prime calcolatrici meccaniche.

La comparsa delle prime calcolatrici risale al principio del settecento. E il fatto che la loro scoperta porti il nome di matematici illustri dimostra, come il progresso di questa disciplina fosse fin da allora strettamente legato alla conquista di validi strumenti di calcolo.

Nepero, Pascal, Liebnitz sono gli inventori delle prime calcolatrici.

La macchina di Pascal, chiamata appunto Pascalina, è particolarmente importante, perchè introduce alcuni fondamentali principi, ai quali si ricollegano tutte le moderne calcolatrici meccaniche (figura 2).

Pascal, al posto delle dieci dita della mano o dei dieci sassolini, usò una ruota con dieci denti; ognuno di questi denti rappresenta un numero. Per indicare il numero basta girare la ruota dentata in corrispondenza di un punto fisso.

La ruota dentata nella figura 2 esprime in questa posizione il numero 7. Le diverse ruote esprimono rispettivamente numeri di unità decine, centinaia, migliaia ecc.; per cui ad es. il numero 234 è ottenuto ruotando di 4 la ruota delle unità, di 3 la ruota delle decine, di 2 la ruota delle centinaia. Questo per quanto riguarda l'impostazione del numero.

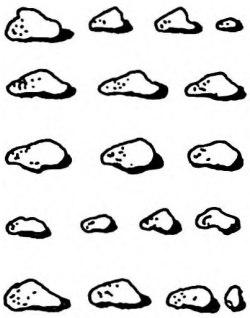
Per i calcoli Pascal ha ideato il riporto automatico della decina. Dopo un giro completo la ruota fa avanzare di uno scatto la ruota di sinistra immediatamente vicina, proprio come si fa in aritmetica addizionando i numeri in colonna.

Il meccanismo del riporto ha una rilevanza importantissima nel campo del calcolo automatico, in quanto lascia compiere alla macchina una funzione che coll'abaco a pallottoliere spettava invece all'operatore.

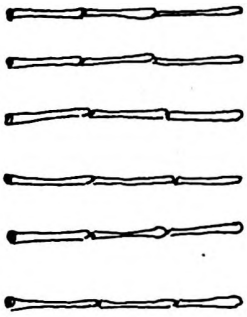
FIGURA 1-1

RUDIMENTALI STRUMENTI DI CALCOLO

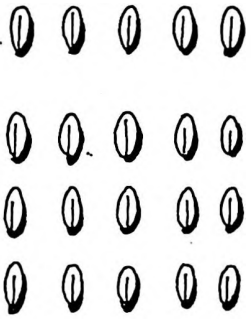
SASSOLINI



CANNUCCE



CHICCHI DI GRANO



ABACO

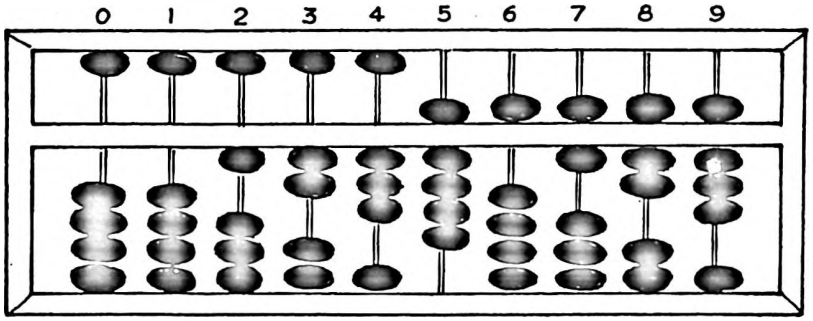
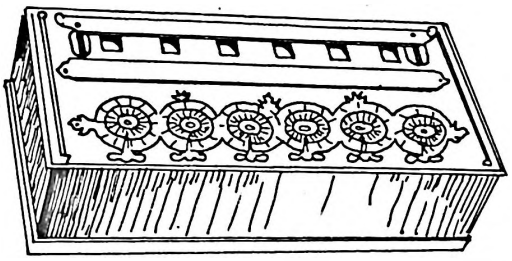
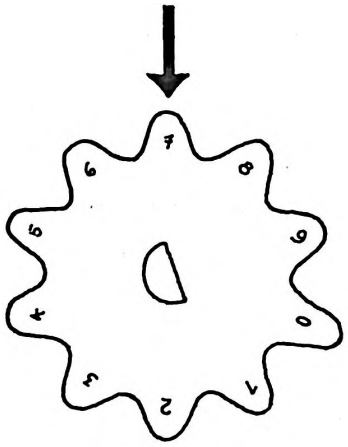
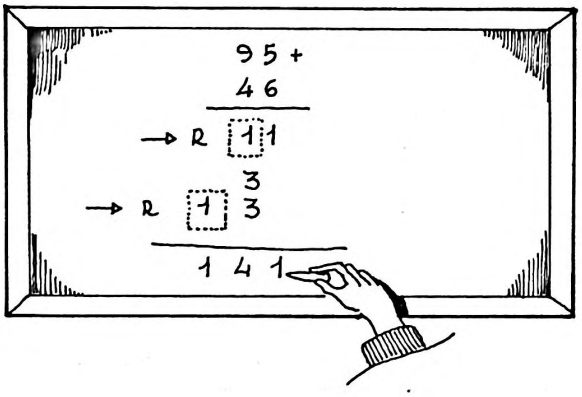


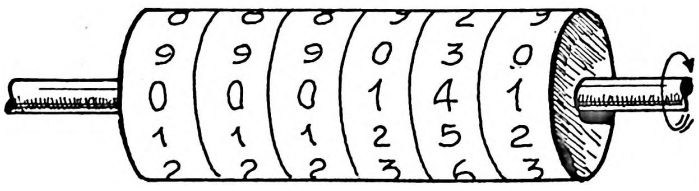
FIGURA 1-2

LA PASCALINA (1642)









Nel 1830 Charles Babbage, professore dell'Università di Cambridge, inventa la prima calcolatrice meccanica completamente automatica.

I suoi tentativi di costituirla sono limitati ad un semplice prototipo, in quanto gli strumenti tecnici dell'epoca sono molto scarsi ed arretrati. La precocità delle sue intuizioni fecero sì che i contemporanei chiamassero la sua macchina col nome di "Babbage Folly". Ciò consente di guardare alla sua opera, come ad un precursore dei tempi moderni. La macchina infatti è dotata di congegni veloci ed interdipendenti che svincolano l'operatore dal lavoro di impostazione e di comando di una operazione successiva ad un'altra. Questi furono in definitiva i principi della calcolatrice automatica di Babbage.

A questo proposito la macchina contiene, sia pure in forma embrionale tutte le componenti fondamentali dei moderni calcolatori: Memoria, Dispositivo di calcolo, Governo, Supporto d'entrata, Supporto d'uscita.

La memoria incamera i dati di base che prima l'operatore era costretto a leggere dal foglio e ad impostare su tastiera.

Il dispositivo di calcolo, esegue tutte e quattro le operazioni, fondandosi sull'applicazione della sola addizione e sottrazione.

Il Governo è l'organo che si sostituisce nel passaggio da una operazione alla successiva, usufruendo dei risultati ottenuti e combinandoli con altri dati secondo un certo programma. La macchina di Babbage è una macchina programmata, cioè fornita di comandi automatici detti istruzioni, che sono registrati in memoria in una zona distinta da quella dei dati e che permettono di "istruire" la macchina sul lavoro da eseguire.

Lo scambio di dati fra la memoria e l'esterno è ottenuto col tramite di cartoncini rettangolari forati. La posizione del foro consente alla macchina di distinguere i diversi caratteri componenti i dati.

Le intuizioni e i progetti di Babbage furono sviluppati una cinquantina di anni più tardi, quando Hollerith, incaricato dal governo americano di eseguire una statistica sulla popolazione, costruì una macchina, molto simile alle moderne macchine a schede perforate, servendosi di applicazioni elettriche. Infatti i caratteri presenti nella scheda di carta sono trasferiti in memoria sotto forma di impulsi elettrici, ottenuti mediante una serie di aghi, che possono, a seconda che è presente o no il foro, pescare in una bacinella di mercurio la quale effettua la chiusura di un circuito.

Il Powers costruisce anch'egli una macchina analoga a quella dell'Hollerith, ma completamente meccanica.

Più tardi verso il 1925, Frederik Rosing Bull progetterà apparecchiature a schede perforate particolarmente geniali ed efficaci per la risoluzione di svariati problemi di contabilità e statistica, che anticipano nel tempo soluzioni tecniche realizzate altrove solo molto più avanti.

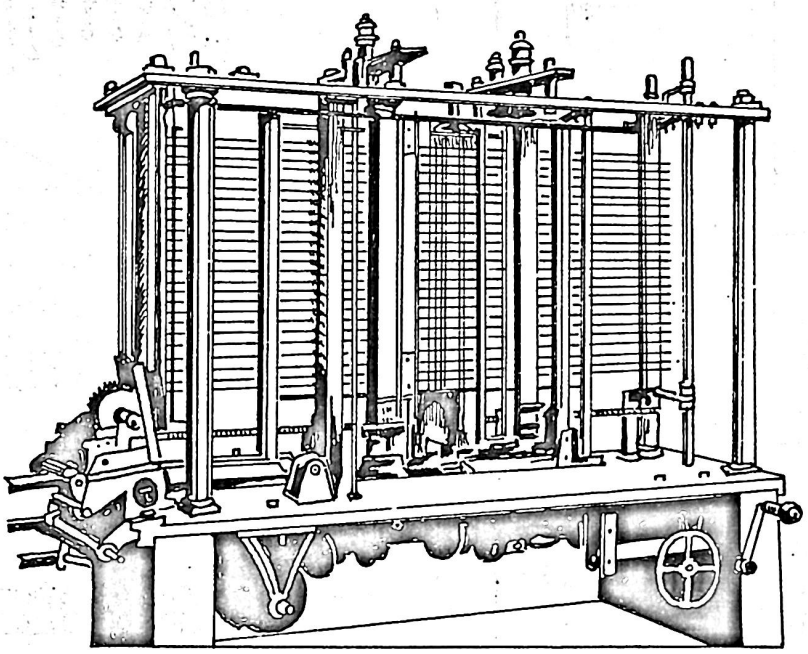
1. 4 La produzione industriale dei calcolatori e degli elaboratori

Dopo la prima guerra mondiale, la necessità di risolvere una enorme quantità di calcoli, e di elaborare una massa cospicua di dati stimola da parte degli operatori economici la domanda di un mercato di calcolatori in grado di sveltire le procedure. Questa richiesta determina la produzione dei calcolatori e degli elaboratori su scala industriale. Col vantaggio di concentrare proficuamente quegli sforzi, che prima erano ritenuti bizzarie di studiosi e scienziati. Un mercato di calcolatori, come qualsiasi altro, per prosperare necessita continuamente dell'apporto di idee nuove, i vecchi "pazzi" diventano i moderni tecnici e ingegneri al servizio della produzione e del progresso.

Parallelamente a questo fatto si deve considerare la possibilità di sfruttamento di nuove energie e nuovi strumenti, la cui funzione sarà determinante nel dare allo sviluppo dei calcolatori e

FIGURA 1-3

MACCHINA BABBAGE



SCHEMA FUNZIONALE DELLA CALCOLATRICE DI BABBAGE

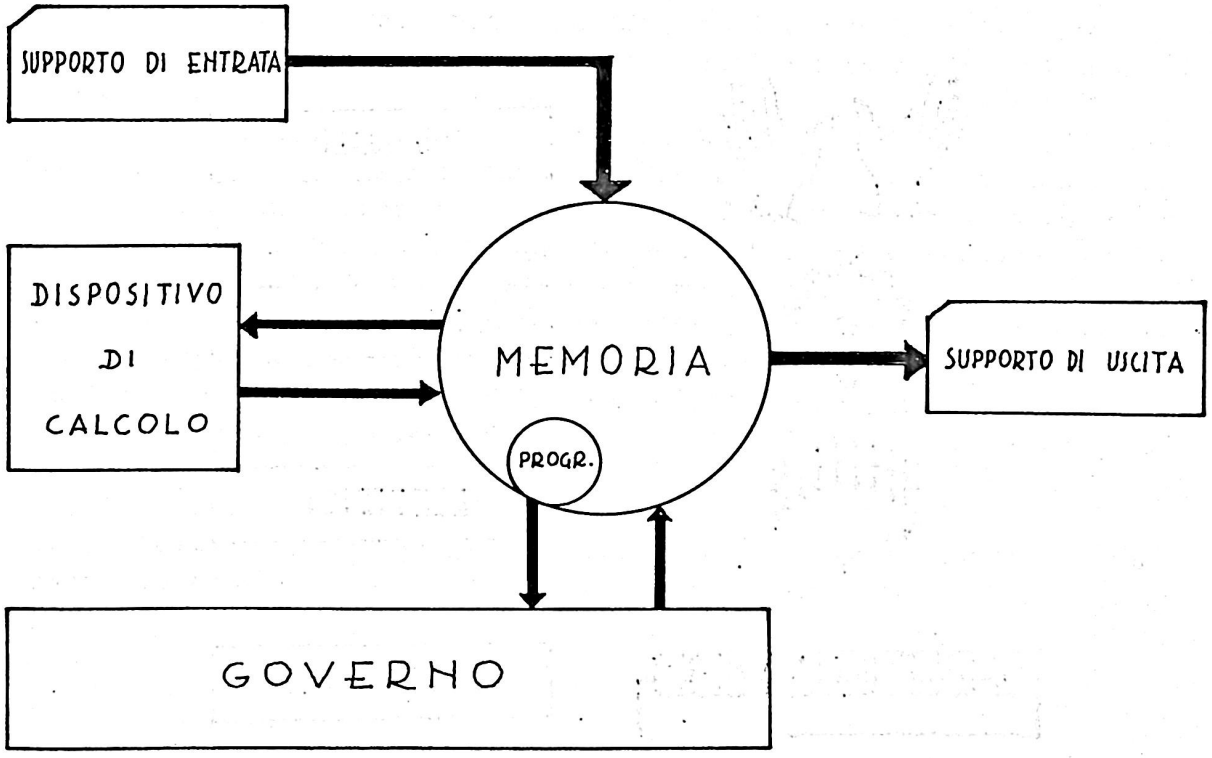
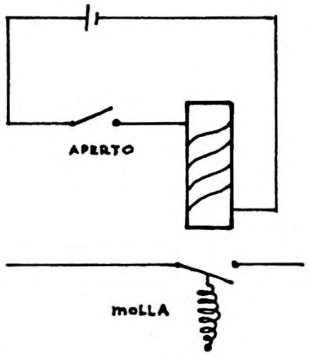
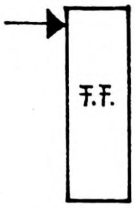
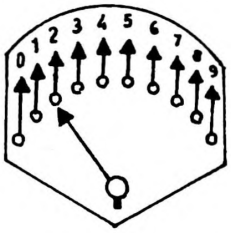


FIGURA 1 - 4

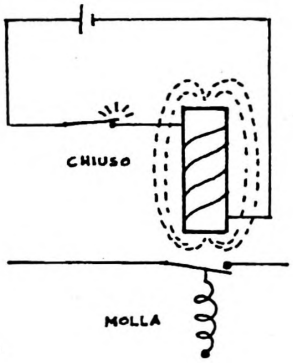
RELAY	FLIP-FLOP	RELAY SELETTORE A 10 POSIZIONI

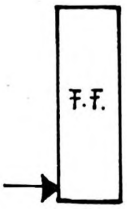


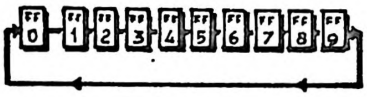




CONTATORE A FLIP-FLOP





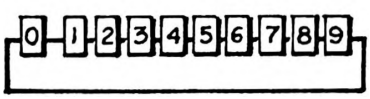


CONTATORI DECIMALI

MANUALI

MECCANICI

ELETTRICI



ELETTRONICI

degli elaboratori una spinta e un movimento addirittura iperbolici. Sulla scia di questa prospettiva si profilano le fisionomie di nuovi tecnici e specialisti intesi a risolvere i problemi connessi all'utilizzazione di queste macchine. Analisti, programmatori, codificatori sono i pionieri di una schiera destinata a moltiplicarsi nell'avvenire.

1.5 Il relay e i componenti elettronici.

La scoperta del relay dà il via ad una utilizzazione affatto nuova per gli strumenti di calcolo. Il suo apporto è infatti decisivo sia per la maggiore velocità sia per un ulteriore sviluppo delle funzioni di governo in un calcolatore.

Il relay tecnicamente realizza la chiusura di un circuito in seguito ad una eccitazione che si propaga attraverso le sue spire, in conseguenza di un impulso elettrico. Tale chiusura determina il passaggio di un impulso in un nuovo conduttore, il quale se collegato ad un altro relay, dà vita ad un altro impulso e così via.

Questi impulsi presi singolarmente possono equivalere ad un comando elementare. Variamente combinandoli danno luogo a comandi complessi che realizzano le diverse funzioni di governo e di calcolo.
Insieme a questi relay si hanno i relay selettori, dove l'energia magnetica è utilizzata per far ruotare una lancetta che, fermandosi su punti distinti di un quadrante, stabilisce di volta in volta contatti diversi. In questo modo il relay selettore sostituisce la "ruota dentata decimale" con l'evidente vantaggio di una maggiore velocità. Il relay selettore è in grado di provvedere all'impostazione e al calcolo di numeri. Il numero 3 è ottenuto facendo partire tre impulsi in un selettore azzerato. L'operazione 3 + 2 = 5 si ottiene avviando altri 2 impulsi sullo stesso selettore.

Quasi tutti i centri meccanografici, che oggi chiamiamo tradizionali in contrapposto ai moderni centri elettronici, utilizzano

il relay come elemento fondamentale di memoria di calcolo e di governo.

Se i relay erano molto più veloci delle ruote meccaniche, molto più veloci dei relay, dovevano presto rivelarsi gli elementi elettronici. Un relay opera approssimativamente in un centesimo di secondo. Tuttavia il tempo degli elementi elettronici è enormemente più basso e raggiunge il livello di milionesimi di secondo o microsecondi. il primo elaboratore elettronico era già in grado di fare 50 addizioni, nel tempo in cui il relay operava una sola commutazione !

Si pensi che i primi componenti elettronici risalgono al 1940 ma grazie ai notevoli sviluppi del radar e della TV essi hanno permesso alle industrie degli elaboratori elettronici di conseguire risultati sorprendenti!

La funzione del componente elettronico è perfettamente analoga a quella del relay. Il componente elettronico è costituito da un circuito operante in un tubo a vuoto (recentemente transistorizzato), il quale può assumere due stati elettrici opposti che sono in grado di commutarsi con una velocità elevatissima.

La commutazione (cambiamento di stato) avviene cambiando i valori delle correnti di alimentazione. A questo meccanismo si dà il nome di flip - flop o trigger o toggle.

Similmente al relay, il flip-flop è utilizzato per funzioni di governo di memorizzazione e totalizzazione. In questo ultimo caso al relay selettore è sostituita una catena di 10 flip flop.

1.6 Gli Elaboratori elettronici

Il primo Elaboratore elettronico viene costituito dall'Università di Pennsylvania nel 1943 e si compone degli elementi elettronici già visti, i quali gli conferiscono un'elevata velocità. Naturalmente esso possiede una memoria per i dati da elaborare, u-

na memoria per le istruzioni di macchina e una potente unità di governo capace di svolgere le operazioni in modo automatico e controllato. L'organo di calcolo è un contatore a elementi di dieci flip-flop, conosciuto come "contatore circolare", perchè i suoi flip-flop sono collegati l'uno con l'altro a catena, in modo che una volta arrivata a 9 ritorna a 0. I numeri sono espressi in forma decimale con tanti impulsi elettronici quanto è il loro valore. Per esempio il numero tre, con tre impulsi, il numero 8 con otto impulsi consecutivi. Fra una catena e la successiva esiste il riporto.

Questo prototipo è destinato ad avviare una produzione successiva, il cui sviluppo sarà addirittura esplosivo: negli ultimi venti anni il numero e i tipi di elaboratori utilizzati nel campo commerciale tecnico e scientifico non si conta più.

Fra questi nel 1959 la soc. Olivetti presenta sul mercato il suo primo elaboratore Elea 9003 di grande potenza destinato soprattutto all'automazione dei servizi di tipo amministrativo e commerciale, solo un anno più tardi la sua produzione si completa nell'Elaboratore Elea 6001, destinato in modo particolare alla risoluzione di problemi tecnici e scientifici.

1.7 Conclusioni

Dopo la scoperta di Biagio Pascal, il progresso del calcolo automatico può dirsi indirizzato al conseguimento dei seguenti obiettivi:

1) Trasferire dall'operatore alla macchina un numero sempre maggiore di funzioni.

2) Rendere l'esecuzione di queste funzioni sempre più veloce e tempestivo.

3) Sottoporre lo svolgimento delle funzioni (almeno le più complesse) a controlli di vario ordine in modo da localizzare gli errori ed avere la sicurezza del procedimento.

4) Realizzare automatismi di tale potenza da consentire lo studio e la risoluzione di problemi assolutamente nuovi e altrimenti impossibili.

Per approfondire meglio l'argomento si pensi alle difficoltà che ha dovuto superare l'uomo per risolvere il problema del trasporto (figura 1.5).

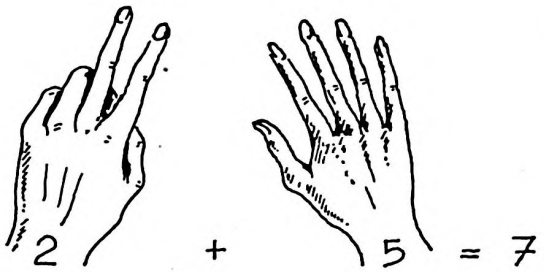
Dapprima si sarà servito unicamente delle proprie forze, questo gli avrà permesso di trasportare quantitativi esigui a distanza molto ravvicinata. Nello stesso modo i matematici si servirono interamente delle loro facoltà mentali, aiutandosi con le dieci dita.

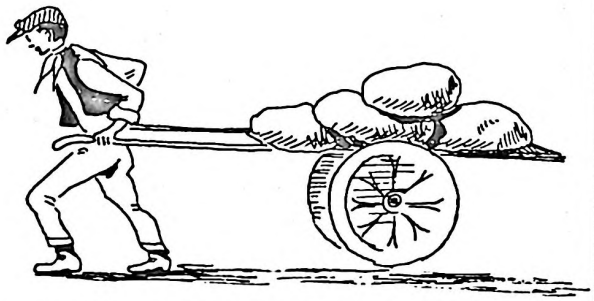
Successivamente l'uomo farà uso di veicoli a trazione, le sue energie restano ugualmente impiegate, ma non così duramente come prima. Per cui gli sarà dato di trasportare carichi più pesanti a distanze più elevate. E' il caso dell'abaco a pallottoliere e di altri sistemi embrionali di calcolo, senz'altro più agevoli dell'uso delle dita.

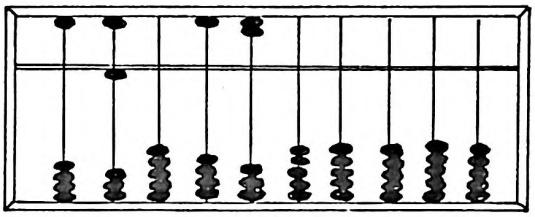
Addomesticati alcuni animali, utilizzata la vela nel campo della navigazione, egli si svincola da un lavoro materiale e oneroso, concentrando le energie in un tenace lavoro di guida manuale. Nel campo del calcolo compaiono le prime macchine da tavolo, dove l'attività umana è impiegata nell'impostazione dei dati nell'uso dello strumento e nella rilevazione dei risultati.

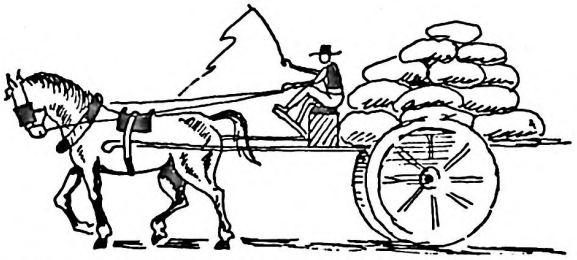
La scoperta del motore a scoppio, la sua utilizzazione nel campo dei veicoli, mettono a disposizione dell'uomo una vasta gamma di automezzi, terrestri, marittimi e aerei. Egli può trasportare enormi quantità da un punto all'altro della terra. E, se vede da questo lato diminuire lo sforzo materiale, da un altro egli deve accrescere le proprie conoscenze tecniche per essere in grado di provvedere alla conduzione dei nuovi mezzi di trasporto. Nel calcolo automatico arrivano le macchine contabili, le fatturatrici, i centri a schede perforate che, permettendo la soluzione di problemi lunghi e laboriosi, trovano nel campo a-

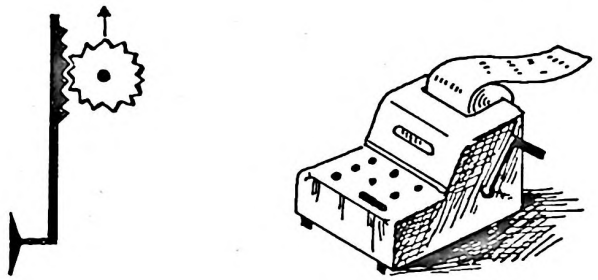
FIGURA 1-5

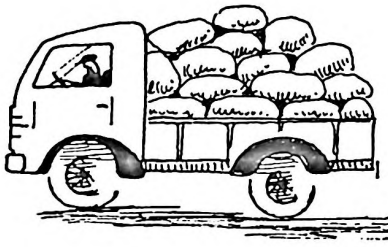


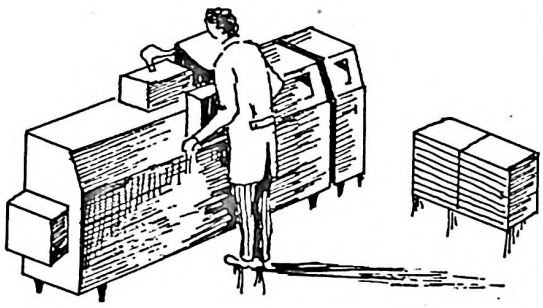


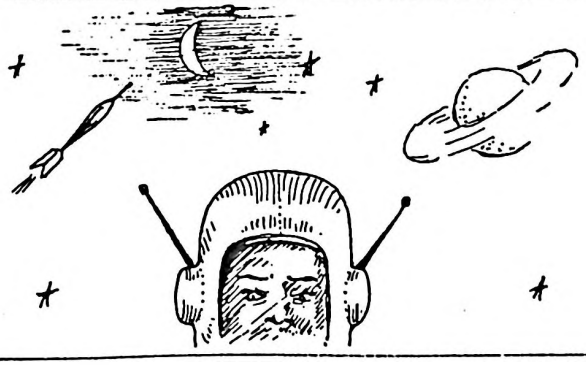


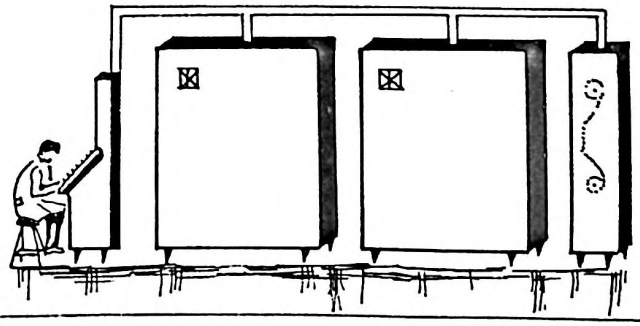












ziendale le più fertili applicazioni. Lavori di contabilità, di fatturazione, di paghe, di statistiche, aggiornamenti di archivi, vengono correttamente e celermente svolti dai centri meccanografici, coadiuvati da altre macchine ausiliarie. E' necessaria l'istruzione e la specializzazione di tecnici capaci di operare su queste nuove macchine.

Il crescente perfezionamento delle tecniche e delle scienze, il moltiplicarsi delle scoperte, mettono a disposizione dell'uomo energie e strumenti che fino a pochi anni addietro erano bizzarrie della fantascienza, pure chimere da vagheggiarsi soltanto nei sogni. I razzi balistici, i missili, le navi spaziali muovono alla conquista e all'esplorazione di mete celesti finora sconosciute. Lo stesso avviene nel calcolo automatico colla costruzione dei moderni elaboratori elettronici. Non solo i vecchi e tradizionali problemi vengono risolti a velocità spaventosamente sorprendenti, ma si consente agli scienziati e ai tecnici l'esplorazione di campi e di discipline assolutamente nuovi. Si pensi alla ricerca operativa, alla programmazione della produzione, ecc. .

E S E R C I Z I

1) Perchè lo sviluppo del calcolo è stato parallelo allo sviluppo della matematica e delle scienze?

2) Quale pregio ha avuto la scoperta di Pascal?

3) Perchè la macchina di Babbage ha le caratteristiche di una macchina automatica?

4) Quando fu scoperto il primo Elaboratore elettronico?

5) Perchè secondo voi un moderno elaboratore è composto di elementi elettronici?

6) Quando fu lanciato sul mercato l'Elea 9003?

II SISTEMI DI NUMERAZIONE

2.1 Il sistema decimale

Quando gli uomini cominciarono a contare, probabilmente fecero uso delle dita. Essi contavano unità per unità aggiungendo un altro dito per rappresentare un'unità in più.

Il sistema decimale di numerazione, sorto appunto in rapporto all'uso primitivo delle dieci dita delle mani, esprime tutti i numeri servendosi di dieci configurazioni, combinate fra di loro secondo regole determinate.

Fin da principio dovette risultare evidente che con dieci dita a disposizione non era possibile esprimere più di dieci numeri e che, se occorreva rappresentare numeri di valore superiore, si doveva fare uso di troppe mani, cioè di troppi individui.

- 4 uomini per rappresentare 34 unità!

Figuriamoci quanti ne occorrevano per cifre più elevate!

Allora si pensò di economizzare il numero degli individui, che evidentemente rendeva il procedimento assai laborioso. Così si designarono le dita di uno come rappresentanti le unità e quelle del compagno, come rappresentanti le decine. In questo modo si poterono esprimere fino a 100 numeri con due soli individui, anzichè dieci.

3 decine + 4 unità

30 + 4 = 34

La necessità di rappresentare i numeri in sistema, si dovette sviluppare come soluzione, che valesse a risparmiare un certo numero di individui nella conta.

Le dieci dita della mano servirono sempre come espressione di base, ma avevano un peso diverso a seconda dell'individuo a cui ap

partenevano.

Quando si cominciarono a rappresentare i numeri graficamente, la necessità di un sistema fu dettata dal fatto di ricordare a memoria un numero limitato di caratteri rappresentativi, scelti come base.

Colla rappresentazione decimale i caratteri da ricordare sono soltanto dieci, che raggruppati in sistema consentono di rappresentare tutti i numeri che si desiderano.

E' evidente che, se per rappresentare ad es. il numero 1841 e il numero 719, occorresse un carattere particolare per ciascuno, non basterebbero 10 anni di studio a ricordare i primi 10.000 numeri! Allo stesso modo che ai primi uomini sarebbero occorsi 1.000 compagni!

Il sistema decimale è una combinazione di caratteri e di posizioni. Ogni posizione contiene uno dei 10 caratteri, però in ogni posizione un carattere ha un peso differente. Cominciando da destra verso sinistra ciascun carattere ha rispettivamente peso 1, 10, 100, 1000, eccetera.

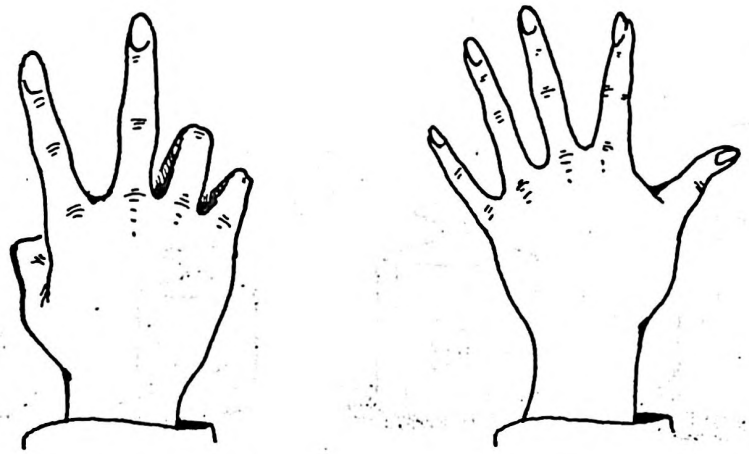
In questo modo il numero 333, vede lo stesso carattere 3 usato 3 volte. Il 3 di estrema destra indica 3 unità, quello di mezzo 3 decine, quello di sinistra 3 centinaia.

Il numero	3	3	3		
Significa	300	+30	+3	=	333

Come si osserva dall'esempio riportato in figura 2-2, il sistema decimale rappresenta i numeri, scomponendoli in potenze di 10 successive (10^0, 10^1, 10^2, 10^3, ecc.) e ponendo nella colonna corrispondente il numero delle potenze aventi lo stesso esponente. Esempio: il numero 4 2 3 5

FIGURA 2-1

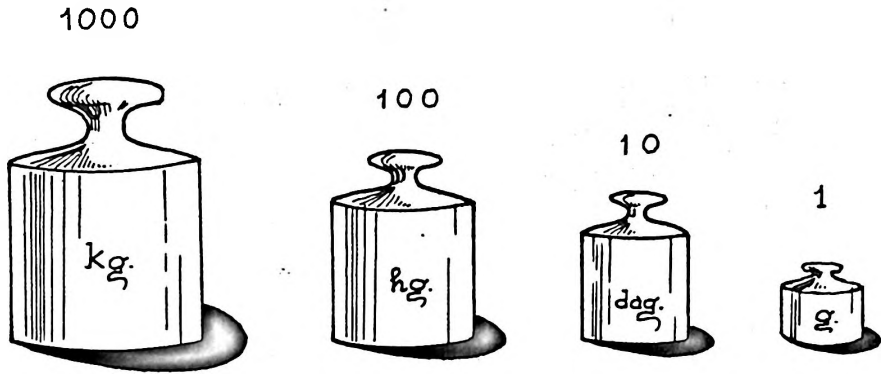
2 + 5 = 7



4 UOMINI PER RAFFIGURARE 34 UNITA'

3 DECINE + 4 UNITA' = 34

FIGURA 2-2 ;



IL NUMERO → 3 3 3

SIGNIFICA → 300 + 30 + 3 = 333

MIGLIAIA 10^3 MODULO 1000	CENTINAIA 10^2 MODULO 100	DECINE 10^1 MODULO 10	UNITA' 10^0 MODULO 1
4	2	3	5

4 × 1000 → 4000

2 × 100 → 200

3 × 10 → 30

5 × 1 → 5

4000 + 200 + 30 + 5 = 4235

$4235 = 10^0 + 10^0 + 10^0 + 10^0 + 10^0 + 10^1 + 10^1 + 10^1 + 10^2 + 10^2 + 10^3 + 10^3 + 10^3 + 10^3 = 5\ (10^0) + 3\ (10^1) + 2\ (10^2) + 4\ (10^3)$;

il numero $803 = 3\ (10^0) + 0\ (10^1) + 8\ (10^2)$

Da quanto detto risulta evidente, che il passaggio da una colonna alla successiva di sinistra avviene ogni volta che il numero supera il valore 9, inteso come 9 unità per la prima colonna, 9 decine per la seconda, 9 centinaia per la terza etc. In tal caso il valore di riporto lo si trasferisce sulla colonna più vicina di sinistra, sommandolo col valore ivi già presente.

2.2 Il sistema binario

Il sistema di numerazione più semplice è quello su base due, il quale si serve di due sole configurazioni di caratteri: 0 (zero), 1 (uno). Questo metodo è abbastanza usato nei calcolatori e negli elaboratori, per il semplice fatto che la maggior parte dei loro componenti sono di natura binaria. Cioè a dire, sono suscettibili di esprimere attraverso il loro stato o la loro posizione, solo due valori differenti.

Come nella numerazione su base 10 il numero di ciascuna colonna indica quante volte viene espressa una certa potenza di 10, così nella numerazione su base 2 ciascuna colonna indica quante volte vi viene espressa una data potenza di 2. Ovviamente, essendo solo due le configurazioni a disposizione, tale numero sarà come minimo 0 (zero), come massimo 1 (uno).

Costruiamo in questo modo i primi 10 numeri.

Espressione decimale		Espressione binaria
0	$= \ldots\ldots\ldots\ldots\ldots\ldots 0x2^0 =$	0 0 0 0
1	$= \ldots\ldots\ldots\ldots\ldots\ldots 1x2^0 =$	0 0 0 1
2	$= \ldots\ldots\ldots\ldots\ldots 1x2^1+0x2^0 =$	0 0 1 0
3	$= \ldots\ldots\ldots\ldots\ldots 1x2^1+1x2^0 =$	0 0 1 1
4	$= \ldots\ldots\ldots\ldots 1x2^2+0x2^1+0x2^0 =$	0 1 0 0
5	$= \ldots\ldots\ldots\ldots 1x2^2+0x2^1+1x2^0 =$	0 1 0 1
6	$= \ldots\ldots\ldots\ldots 1x2^2+1x2^1+0x2^0 =$	0 1 1 0
7	$= \ldots\ldots\ldots\ldots 1x2^2+1x2^1+1x2^0 =$	0 1 1 1
8	$= \ldots\ldots\ldots 1x2^3+0x2^2+0x2^1+0x2^0 =$	1 0 0 0
9	$= \ldots\ldots\ldots 1x2^3+0x2^2+0x2^1+1x2^0 =$	1 0 0 1
10	$= \ldots\ldots\ldots 1x2^3+0x2^2+1x2^1+0x2^0 =$	1 0 1 0

o p p u r e

Da decimale		a		binaria
0	=		=	0 0 0 0
1	=	1 = 0000 + 1	=	0 0 0 1
2	=	1 + 1 = 0001 + 0001	=	0 0 1 0
3	=	2 + 1 = 0010 + 0001	=	0 0 1 1
4	=	3 + 1 = 0011 + 0001	=	0 1 0 0
5	=	4 + 1 = 0100 + 0001	=	0 1 0 1
6	=	5 + 1 = 0101 + 0001	=	0 1 1 0
7	=	6 + 1 = 0110 + 0001	=	0 1 1 1
8	=	7 + 1 = 0111 + 0001	=	1 0 0 0
9	=	8 + 1 = 1000 + 0001	=	1 0 0 1
10	=	9 + 1 = 1001 + 0001	=	1 0 1 0

Questi numeri ci mostrano che il riporto avviene ogni qualvolta una colonna supera il valore 1 (cioè il carattere che ha valore più alto) e che, a seguito di ciò detta colonna passa a 0 mentre si somma 1 sulla colonna successiva di sinistra.

Proviamo ad addizionare 5 + 9

Espressione Decimale	Espressione Binaria
5 +	0 1 0 1
9	1 0 0 1
(1) 4	(1) 0
	1
	1
	1
1 4	1 1 1 0

---

6 +	0 1 1 0
3	0 0 1 1
9	1
	(1)0
	(1)0
	1
9	1 0 0 1

Il seguente prospetto dimostra chiaramente questo punto: minore è il numero di caratteri che figurano in una colonna, maggiore è il numero di colonna che necessitano per rappresentare una determinata quantità. Il numero nove per esempio nella numerazione unitaria (un solo carattere per colonna) necessita di ben dieci colonne, nella numerazione binaria di quattro colonne, nella numerazione decimale di una sola colonna.

Numerazione unitaria	Numerazione binaria	Numerazione decimale
1	0000	0
11	0001	1
111	0010	2
1111	0011	3
11111	0100	4
111111	0101	5
1111111	0110	6
11111111	0111	7
111111111	1000	8
1111111111	1001	9
11111111111	1010	10
111111111111	1011	11
1111111111111	1100	12

2.3 Sistema decimale o sistema binario

Il sistema binario applicato ad un elaboratore presenta dei vantaggi "interni" che consentono di semplificare le operazioni di calcolo e quelle di controllo, rendendo anche più rapida la procedura di esecuzione.

Tuttavia a questi vantaggi "interni" si contrappongono delle difficoltà di natura pratica.

I dati di qualsiasi problema vengono infatti somministrati agli elaboratori sotto forma decimale, altrettanto dicasi per i risultati che l'elaboratore deve restituirci.

Sarebbe interessante infatti notare la reazione di un uomo d'affari che si veda consegnare un assegno allo sportello di una banca, che anzichè avere scritto ad es. £. 1088 avesse la corrispondente espressione binaria

£. 1000 1000 000 !

Il sistema binario non può essere usato al di fuori dell'elaboratore, in quanto non è affatto di uso comune. Occorre quindi, per gli elaboratori che utilizzano il sistema binario, intercalare dei convertitori che, per i dati in entrata, codifichino i numeri decimali in numeri binari e viceversa per i dati in uscita provvedano a decodificare i numeri binari in numeri decimali.(Figura 2 - 3)

Purtroppo questi procedimenti di conversione sono quanto mai disagevoli ed è questa la ragione per cui molti calcolatori ed elaboratori preferiscono rinunciare ai vantaggi "interni" derivanti da un sistema di numerazione binaria e adottare il sistema decimale.

E S E R C I Z I

1) Cosa s'intende per sistema di numerazione decimale?

2) Cosa s'intende per sistema di numerazione binaria?

3) Esprimete un binario i seguenti numeri:

   20, 25, 11, 42, 64

4) Sommate in binario le corrispondenti espressioni:

   14 + 8 . . . . . .

   22 + 12 . . . . . .

   15 + 9 . . . . . .

5) Fate la corrispondente sottrazione binaria delle espressioni seguenti:

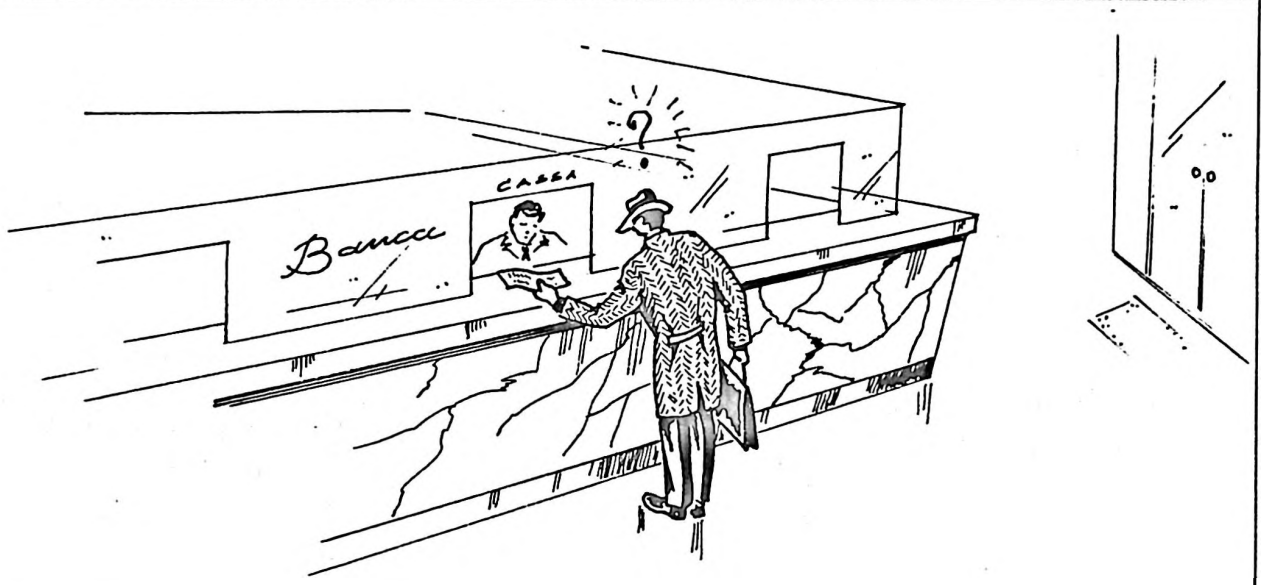
   18 - 7; 22 - 4; 31 - 10.

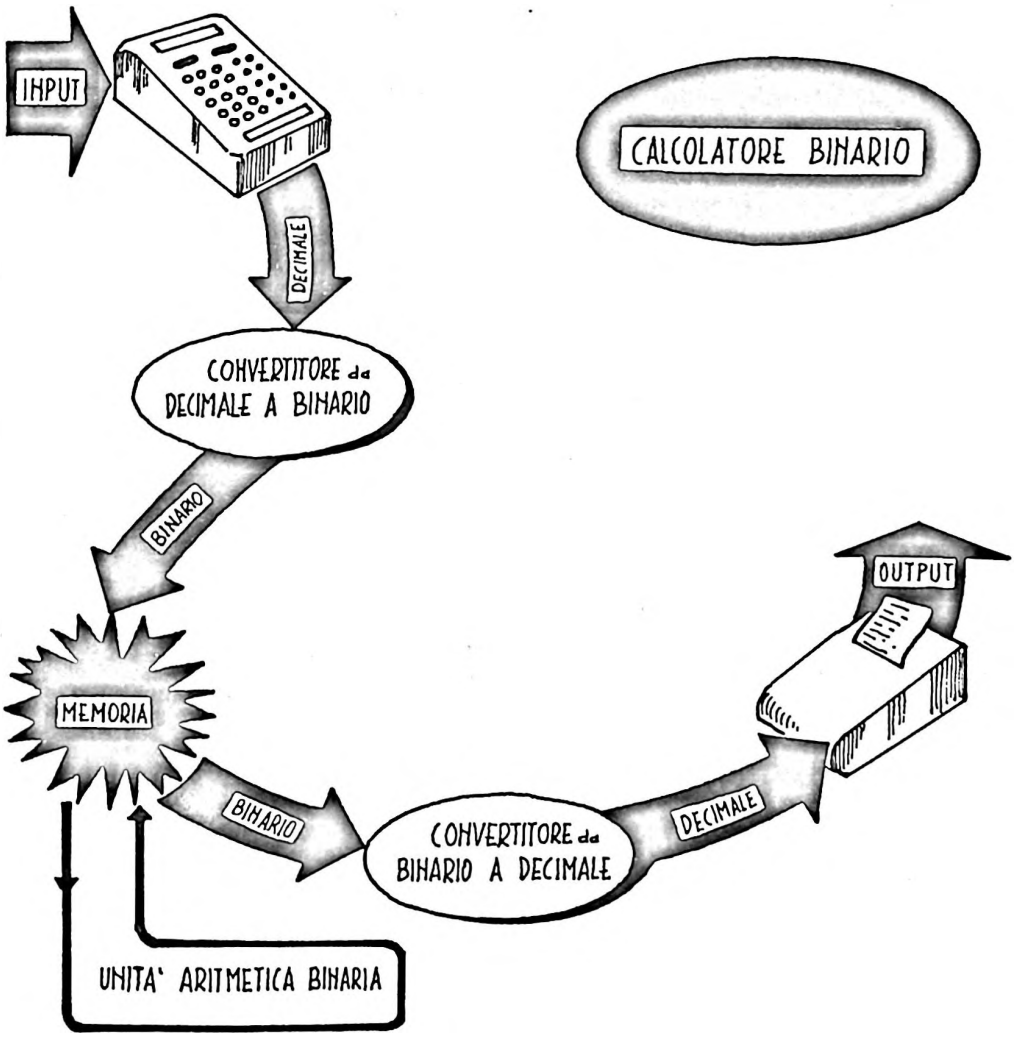
6) Perchè in definitiva un sistema binario è poco agevole per un elaboratore?

7) Quale relazione esiste in un sistema di numerazione tra configurazioni di base e cifre che compongono un numero?

FIGURA 2-3

UN ASSEGNO PER £. 1088 (DECIMALE) SCRITTO IN BINARIO 1000 1000 000





CALCOLATORE BINARIO

III ELEMENTI DELL'ALGEBRA DI BOOLE

3.1 Generalità

Dicesi variabile binaria o booleana, una variabile capace di assumere solo due valori. Si designano normalmente questi due valori con 0 e 1, oppure con NO e SI, o ancora con "falso" e "vero" ecc. A ogni variabile binaria x si fa corrispondere la variabile binaria $\overline{x}$ detta variabile inversa o complementare, tale che:

- se : $x = 0$, $\overline{x} = 1$;
- se : $x = 1$, $\overline{x} = 0$.

Inoltre sarà : $(\overline{x}) = x$.

Esempio di quantità Booleana:

$$A = B, \quad A \neq B$$

infatti una delle due proposizioni è inversa o complementare dell'altra.

Più generalmente, una proposizione logica può essere vera o falsa : questo è un fenomeno booleano.

Si può dare la rappresentazione grafica fig. (3-1,a)

Come si vede la parte tratteggiata corrisponde ad x, il resto del quadrato a $\overline{x}$.

x rappresenta l'appartenenza di un punto del quadrato al cerchio.

3.2 Addizione o Riunione (simbolo + oppure U)

Siano x ed y due variabili booleane , si definisce la loro somma : $x + y$ ($x \cup y$) tale che,

se $x = 0$, $y = 0$ $\quad x + y = 0 \quad (0 + 0 = 0)$;

se $x = 0$, $y = 1$ $\quad x + y = 1 \quad (0 + 1 = 1)$;

se $x = 1$, $y = 0$ $\quad x + y = 1 \quad (1 + 0 = 1)$;

se $x = 1$, $y = 1$ $\quad x + y = 1 \quad (1 + 1 = 1)$.

Dalla tavola si deducono le seguenti proprietà:

$x + y = y + x$;

$x + 0 = x$

$x + 1 = 1$

$x + x = x$

$x + \overline{x} = 1$

Definiamo riunione di due variabili booleane x U y quella nuova variabile z, che presenta le caratteristiche (dette anche predicati) "o" di x "o" di y.

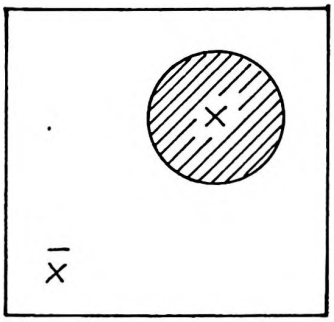
La congiunzione "O" esprime benissimo l'addizione booleana:

- proposizione x = io compro un vestito rosso;
- proposizione y = io compro un vestito verde;
- proporzione z = x U y: io compro un vestito rosso "o" compro un vestito verde.

L'addizione binaria può essere rappresentata da due interruttori disposti in parallelo (Figura 3-1, b), oppure graficamente colla figura (3,1,c), dove la parte tratteggiata rappresenta l'unione della variabile x colla variabile y.

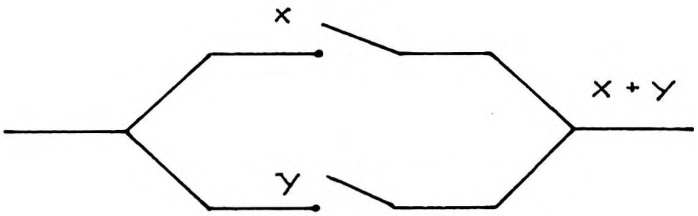
In tal senso l'addizione può essere estesa a più variabili booleane.

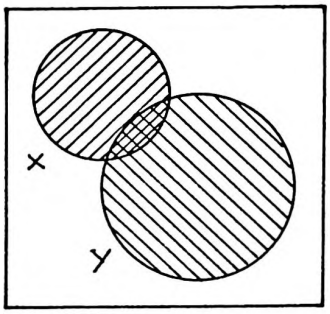
FIGURA 3 - 1



a

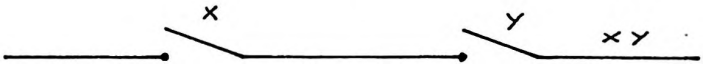
b

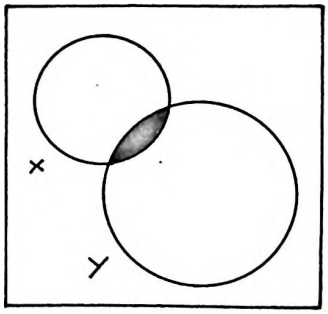




c

d





e

3.3 Moltiplicazione o intersezione (simbolo x o $\cap$)

Siano x e y due variabili booleane si definisce il loro prodotto

x y ($x \cap y$) tale che:

x = 0	y = 0	x y = 0	(0 x 0 = 0)
x = 0	y = 1	x y = 0	(0 x 1 = 0)
x = 1	y = 0	x y = 0	(1 x 0 = 0)
x = 1	y = 1	x y = 1	(1 x 1 = 1)

si deducono le proprietà :

x y =	y x
x 0 =	0
x 1 =	x
x =	x
$x \bar{x}$ =	0

Definiamo "intersezione" di due variabili booleane $x \cap y$ quella variabile z che presenta le caratteristiche (dette anche predicati) "e" di x "e" di y.

Si esprime questa moltiplicazione colla congiunzione "e".

Le due proposizioni "io diventerò ricco e non fumerò più", legate da due eventualità necessarie, traducono l'operazione booleana x $\cap$ y.

Due interruttori figura (3,1,d) collegati in serie schematizzano la moltiplicazione binaria.

Spesso si omette il simbolo "moltiplicato" per designare il prodotto di x y si scriverà x y. Graficamente l'intersezione è rappresentata dalla figura (3,1,e) per la parte tratteggiata.

L'addizione e la moltiplicazione sono delle operazioni associative, commutative e distributive. Si avrà pertanto:

$$(x+y) + z = x + (y+z) = x + y + z = y + x + z$$

$$(x \cdot y)\, z \quad = x \quad (yz\ ) = xyz = yxz$$

$$x\,(y+z) \quad = xy+xz$$

Si dimostrano alcune identità rilevanti :

$$x + xy = \quad x\,(1 + y) = x\ 1 = x$$

infatti in ogni caso sia che sia $y = 0$ oppure $y = 1$

$1 + y$ è sempre 1

$$x\,(x+y) = x \cdot x + x.\ y = x + x\,y = x\,(1+y) = x.\ 1 = x$$

Infatti $x \cdot x$ è sempre x; perchè, se x è 0, il suo prodotto booleano è 0; se x e 1, il suo prodotto booleano è 1.

$$(\overline{x} + y) \ = \ x\,\overline{x} + xy = 0 + xy = xy$$

3.4 Inversione d'una somma o di un prodotto.

Rappresentiamo l'inverso d'una somma e di un prodotto :

x	y	$\bar{x}$	$\bar{y}$	x+y	xy	$\overline{x+y}$	$\overline{y}$	$\overline{x+y}$	$\overline{xy}$
0	0	1	1	0	0	1	1	1	1
0	1	1	0	1	0	0	1	1	0
1	0	0	1	1	0	0	1	1	0
1	1	0	0	1	1	0	0	0	0

$\overline{x+y} = \bar{x}.\bar{y}$: il complemento d'una somma è il prodotto dei complementi;

$\overline{xy} = \bar{x}+\bar{y}$: il complemento di un prodotto è la somma dei complementi.

La regola delle inversioni permette di trovare altre rilevanti proprietà delle espressioni booleane:

$$x + \bar{x}\,y = \overline{(\overline{x + x\,y})} = \overline{\bar{x} + \overline{xy}} = \overline{\bar{x}.\ \overline{(x+y)}} =$$

$$= \overline{\overline{xx} + \bar{x}\,\bar{\bar{y}}} = \overline{\bar{x}\,\bar{\bar{y}}} = x + y\ .$$

E S E R C I Z I

1) Cos'è una variabile booleana?

2) Cos'è l'operazione di riunione?

3) Cos'è l'operazione di intersezione?

4) Quanto risulta: $1 \cap 0$, $1 \cup 1$, $1 \wedge 0$, $1 \vee 0$, $\bar{0} \cap 0$?

5) Quanto fa $x(x \cap y)$?

IV IL LINGUAGGIO DEGLI ELABORATORI

4.1 I linguaggi in generale

Il linguaggio corrente, che si esprime mediante la "parola", pur essendo il più usato, non è che una delle tante forme attraverso le quali è possíbile "comunicare".

Vi sono diversi tipi di linguaggio: il linguaggio parlato, il linguaggio scritto, il linguaggio figurato, il linguaggio ermetico.

La forma del linguaggio è un fatto puramente convenzionale. Tale convenzione è costituita da simboli che possono essere fonetici (vocali e consonanti parlate) o alfabetici (vocali e consonanti scritte), grafici (ideogrammi, geroglifici) o luminosi (segnali con lampade, specchi) o elettriche, elettromagnetiche etc.

Traduttore è qualsiasi strumento in grado di comunicare il "significato" di un linguaggio, in un linguaggio differente. Tanto meno un linguaggio è intelleggibile tanto più necessita di essere tradotto.

Anche le macchine possono avere un linguaggio. La radio, la televisione, il giradischi, hanno un loro linguaggio (espresso sotto forma di onde elettromagnetiche). Ma poichè tali simboli (che sono variazioni di onde) non sono di per sè intelleggibili all'uomo, si inseriscono dei traduttori.

Sembra strano, eppure fra lo speaker della radio e colui che ascolta ci sono "due traduttori". Il primo trasforma le onde sonore della voce in onde elettromagnetiche, il secondo effettua l'operazione inversa. Le comunicazioni fra la stazione emittente e quella ricevente avvengono, per migliaia di chilometri di distanza, sotto forma di "onde elettromagnetiche".

Per fare un altro esempio il microfono e l'altoparlante sono esattamente due "traduttori".

I simboli elementari di un linguaggio in qualunque forma esso venga espresso sono costituiti da caratteri. I caratteri di per sè non dicono nulla, hanno una funzione tipicamente strumentale. Il carattere non sarà mai una parola, per quanto esistono parole di un solo carattere.

Infatti la parola è quell'insieme di caratteri che ha un preciso e specifico significato.

"Roma" insieme dei caratteri R, O, M, A significa la città capitale d'Italia; gli stessi caratteri rovesciati significano sia pure in forma tronca "amore".

Chiameremo invece informazione l'insieme di parole sia alfabetiche, sia numeriche, necessarie a definire qualsiasi fenomeno suscettibile di essere elaborato.

Nel campo aziendale specialmente, le informazioni sono elemento di assoluta rilevanza ai fini della risoluzione dei problemi di cui esse stesse costituiscono il punto di partenza e quello d'arrivo. Dalle informazioni rilevate all'orologio di fabbrica col concorso delle informazioni anagrafiche, si arriva al computo delle paghe degli operai. Dagli ordini evasi, nascono i buoni d'uscita di magazzino, le fatture, i mandati di pagamento, le tratte, gli addebiti di conto corrente. E' una catena di informazioni che non sembra aver fine e senza le quali ciascuna azienda perderebbe fisionomia.

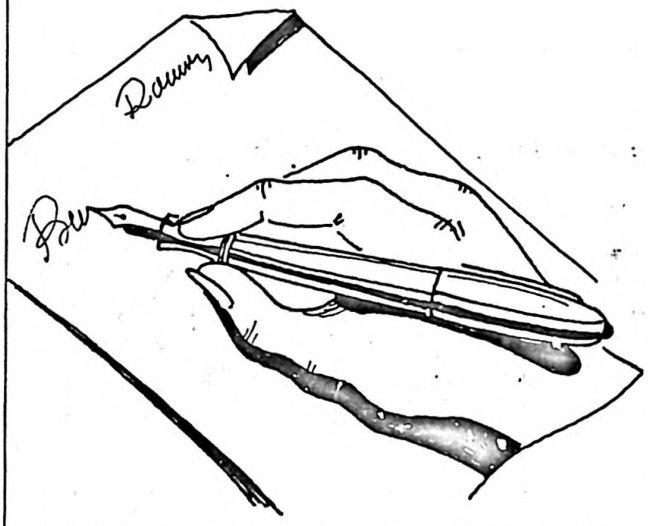
4.2 Il

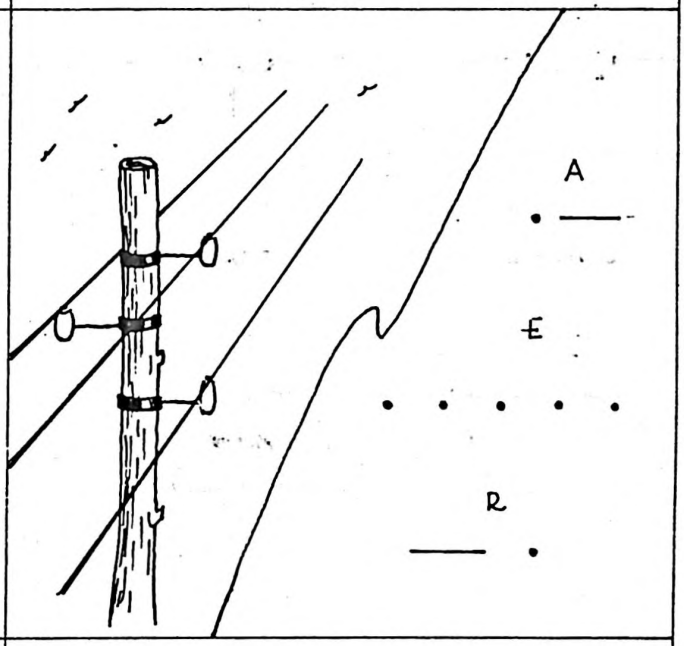
Bit, abbreviazione di "binary-digit", è qualunque variabile binaria, suscettibile di assumere due soli valori chiamati 1, 0.

Il Bit può essere realizzato in diversi modi. Sono infatti tantissimi gli elementi che ci consentono di farlo, alcuni anche banalissimi.

FIGURA 4 - 1

TIPI DI LINGUAGGIO





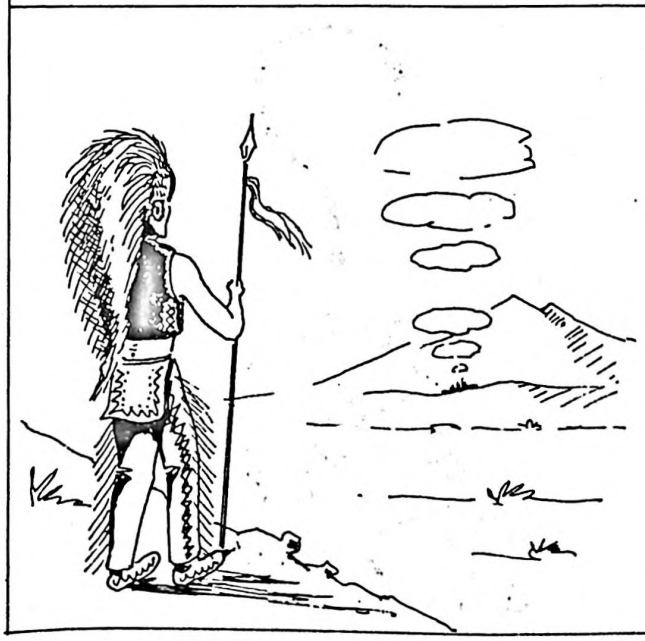
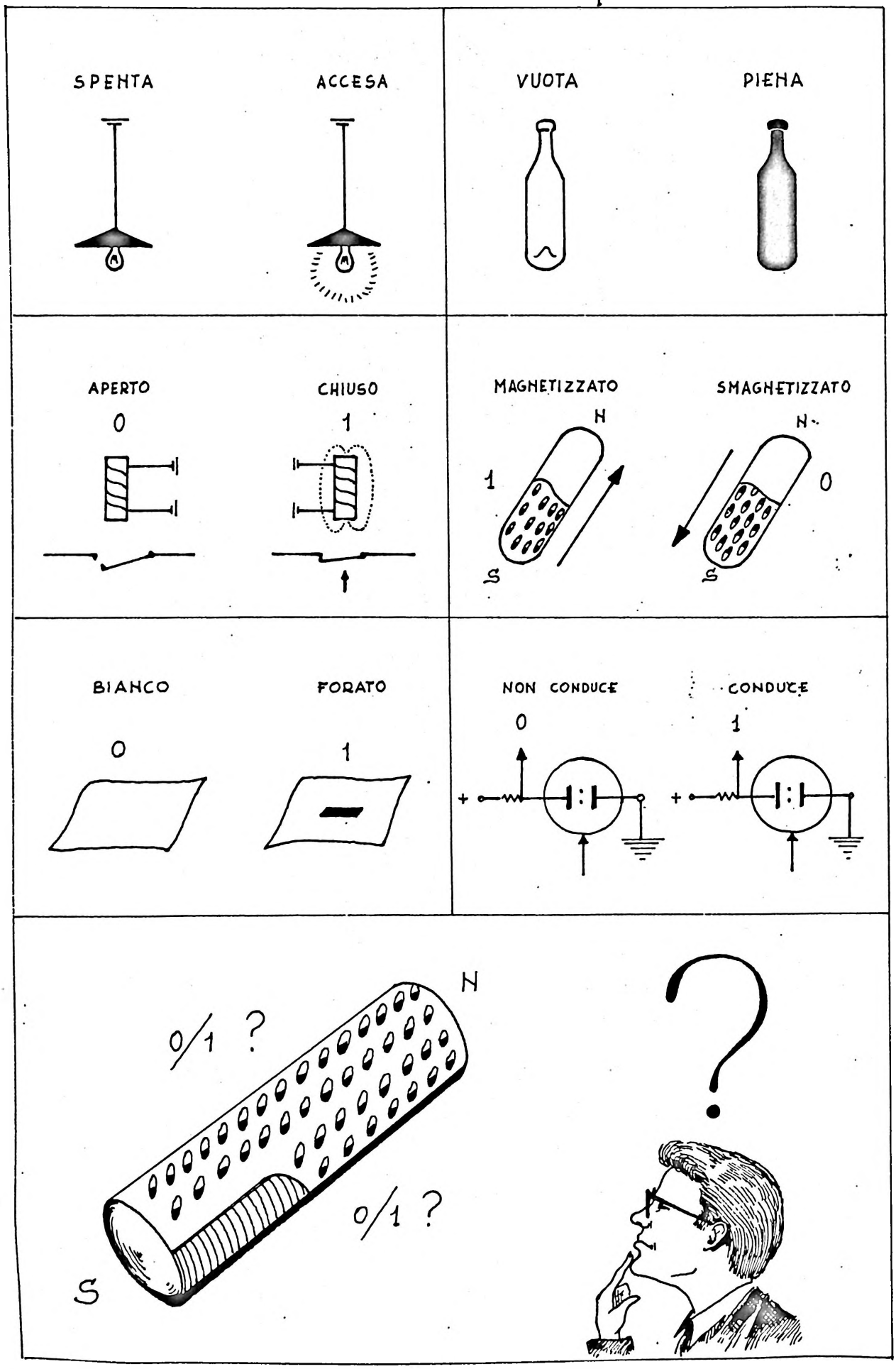


FIGURA 4-2


SPENTA
ACCESA
VUOTA
PIENA
APERTO
0
CHIUSO
1
MAGNETIZZATO
N
1
S
SMAGNETIZZATO
N
0
S
BIANCO
0
FORATO
1
NON CONDUCE
0
CONDUCE
1
+
+
N
0/1 ?
0/1 ?
S

Si pensi ad esempio ad una lampada. Se le diamo valore 0, quando è spenta, 1 quando è accesa essa è in grado di rappresentare perfettamente un bit.

Una bottiglia può essere un bit. Infatti essa può benissimo assumere il valore 0 quando è vuota e 1 quando è piena.

Un pezzo di carta può diventare un bit: gli si dà valore 0, quando è integro, 1 quando è perforato.

Anche un circuito commutato, per esempio da un relay, può costituire un bit: si chiama 0 il circuito aperto, 1 il circuito chiuso.

Così lo stato magnetico di un indotto, quando le particelle sono orientate a nord rappresentano il valore 1 viceversa, quando sono orientate a sud valore 0.

Un tubo a vuoto può assumere due stati diversi, condurre corrente oppure no. Al primo stato possiamo dare valore 1 al secondo valore 0.

Il problema di realizzare un bit è connesso al fatto che l'elemento deve avere delle proprietà "binarie" cioè non deve presentare "zone grige" intermedie, tali da suscitare dubbi sulla attribuzione del valore 0 oppure 1.

Prendiamo ad esempio lo stato magnetico di un indotto: se le sue particelle risultassero orientate in un senso intermedio fra Nord e Sud quale valore attribuire allo stato di questo indotto? In questo caso evidentemente non saremmo di fronte ad un elemento capace di rappresentare un bit.

4.3 I bit e i caratteri

I caratteri possono essere rappresentati mediante "bit".

Con un solo bit possiamo rappresentare due soli caratteri: uno corrispondente allo stato 0, l'altro corrispondente allo stato 1.

Con due bit invece otteniamo le seguenti combinazioni corrispondenti a quattro caratteri:

0 0

0 1

1 1

1 0

Con tre bit, disponendo di un numero doppio di combinazioni rispetto alle precedenti, i caratteri passano a 8:

0 0 0

0 0 1

0 1 1

1 1 1

1 0 0

1 1 0

1 0 1

0 1 0

Aumentando di una unità il numero dei bit a disposizione, raddoppia il numero delle combinazioni e quindi il numero di caratteri che si possono rappresentare.

Con	4 bit	otteniamo	16 caratteri
"	5 "	"	32 "
"	6 "	"	64 "

Dati r bit a disposizione si possono quindi rappresentare n caratteri secondo la formula

$$n = 2^r$$

4.4 Scelta delle configurazioni

Supponiamo di prendere in considerazione questa combinazione di bit: 0 1 0 0 1 1, esiste una ragione particolare in base alla quale questa combinazione deve significare per esempio C piuttosto che 9 o R?

Molto spesso la scelta delle configurazioni da far corrisponder a determinate combinazioni binarie è opzionale, cioè facoltativa, restando chiaro che una volta scelta quella combinazione per quel carattere occorre sempre rispettarla.

Tuttavia (specialmente per i numeri) i tecnici sono soliti studiare delle correlazioni particolari, secondo le quali, posto 1 = 00 0001, per esempio 2 non può che essere 00 0010. I criteri che ispirano i tecnici fanno riferimento i vantaggi pratici nei congegni di calcolo e nella composizione dei circuiti che entreranno a costituire un determinato Elaboratore.

Si chiama codice di macchina quella particolare corrispondenza che esiste fra le combinazioni binarie e i caratteri. Avremo pertanto il codice Elea 9003, il codice Elea 6001 ecc.

4.5 Registrazione e lettura dei caratteri

Supponiamo di disporre di una serie di 6 bit e di volerli configurare in modo da rappresentare per esempio (secondo il codice Elea 9003) il carattere ϕ (fi) = 0 1 0 0 0 0

Si porterà a 0 il bit F, a 1 il bit E, a 0 tutti gli altri.

L'operazione che si è dovuta effettuare per rappresentare questo carattere si chiama registrazione o "scrittura".

Se invece esaminassi la configurazione dei 6 bit per conoscere ciò che vi è espresso, effettuerei una operazione detta appunto "lettura".

4.6 Serie e parallelo

Supponiamo di osservare sei soldati; se essi marciano uno dietro l'altro in colonna, diciamo che la loro disposizione è in "serie" Se invece dovessero marciare in riga per cui tutti e sei procedono frontalmente, diciamo che la loro disposizione è in "parallelo".

Analogamente può accadere per la lettura e la registrazione di un carattere. Gli impulsi, in grado di influenzare magneticamente i bit nel modo dovuto per rappresentare quel determinato carattere, possono partire uno dopo l'altro e allora avremo la registrazione (caso inverso, lettura) in serie; oppure contemporaneamente, in tal caso avremo la registrazione (lettura) in parallelo (figura 4.3).

Normalmente in tutti gli elaboratori i singoli caratteri si leggono e si registrano in parallelo, mentre invece la parola per alcuni è letta in parallelo, per altri in serie.

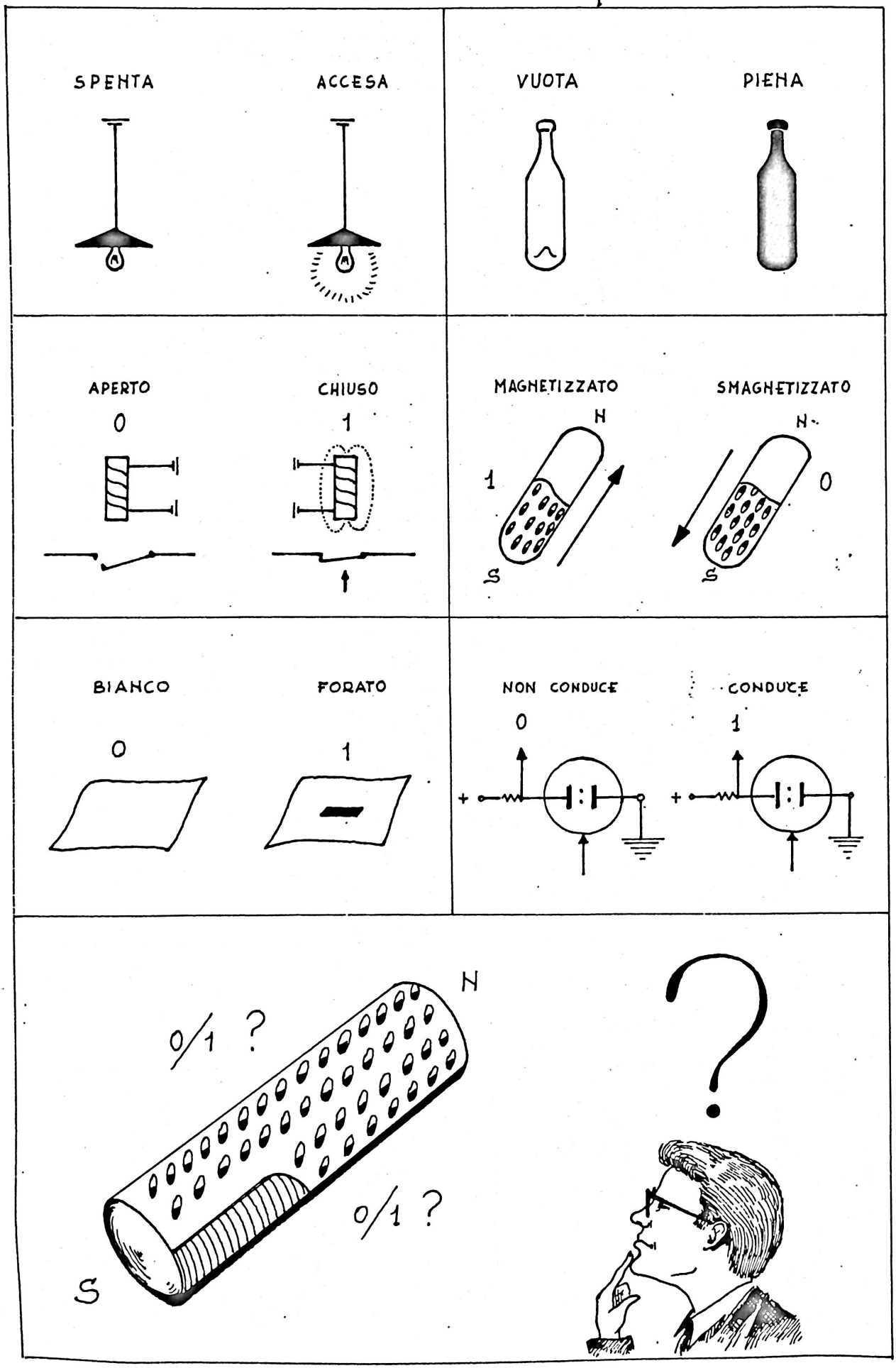
Ovviamente, perchè una parola possa essere letta o registrata in parallelo, occorre che il numero di caratteri che la compongono sia fisso per un determinato elaboratore.

Per alcuni elaboratori la parola è di 10 caratteri, per altri di 6, se la numerazione è decimale, in caso di numerazione binaria la parola sarà ovviamente più lunga cioè 30/40 caratteri.

4.7 Il bit di controllo o check

Una caratteristica fondamentale dei componenti elettronici è quella di procedere a velocità elevatissime. Evidentemente potrebbe

FIGURA 4-2


SPENTA
ACCESA
VUOTA
PIENA
APERTO
0
CHIUSO
1
MAGNETIZZATO
N
1
S
SMAGNETIZZATO
N
0
S
BIANCO
0
FORATO
1
NON CONDUCE
0
CONDUCE
1
+
+
N
0/1 ?
0/1 ?
S

accadere che durante la registrazione e la conseguente lettura di caratteri un bit non acquisti la configurazione pertinente. In tal caso ogni carattere è sottoposto a controlli in diversi punti dell'elaboratore.

Fondamentale è il controllo di disparità (che in alcuni elaboratori è di "parità").

Questo tipo di controllo effettuato da opportuni organi della macchina verifica che il numero dei bit accesi presenti in un carattere sia sempre dispari (oppure pari, se il controllo è di parità).

Perchè questo sia possibile, occorre che, al momento della registrazione del carattere, un bit supplementare di servizio, detto bit check o bit di controllo, intervenga a configurare il carattere con un numero sempre dispari di bit accesi.

La regola consiste in questo: se la configurazione del carattere ha già di per sè un numero dispari di valori 1, il bit di controllo passa a 0; se il carattere ha un numero pari di valori 1, il bit di controllo passa ad 1.

Esempio :

carattere	senza bit di disparità	0 1 0 0 1 0
"	con " " "	0 1 0 0 1 0 1
"	senza " " "	0 0 1 1 1 0
"	con " " "	0 0 1 1 1 0 0

E S E R C I Z I

1) Cos'è un bit?

2) Cos'è un carattere?

3) Cos'è una parola?

4) In che modo un bit può servire a rappresentare un carattere?

5) Esempi di variabili che realizzano un bit.

6) Cosa occorre che possieda un elemento per rappresentare efficientemente un bit?

7) Cosa significa registrare un carattere?

8) Quali sono i modi con cui possiamo leggere o registrare un carattere?

9) Quali sono i modi con cui possiamo leggere o registrare una parola?

10) Cos'è un'informazione?

11) A cosa serve il bit di disparità?

12) Come si comporta il bit di disparità in un determinato carattere?

P A R T E S E C O N D A

L'ELABORATORE E I SUOI ORGANI

I SISTEMA INTEGRATO

1.1 Distinzione fra Calcolatore e Elaboratore

Molto spesso si è soliti confondere impropriamente due termini simili fra di loro, ma con sfumature diverse. I testi infatti parlano indifferentemente di Calcolatori e di Elaboratori, tuttavia è bene qui sottolineare la differenza che li separa.

Posto che il calcolo fa riferimento ad operazioni aritmetiche e algebriche, eseguite su quantità numeriche, Calcolatore sarà qualsiasi strumento idoneo a risolvere automaticamente tali funzioni.

Ma i grandi complessi elettronici non si limitano soltanto a ciò. Il loro compito è assai più vasto; alla base delle loro risoluzioni non sta soltanto il numero, ma l'informazione, precedentemente definita (parte 1^. 4.1) come l'insieme di dati numerici e alfabetici che servono a esprimere un determinato fenomeno. Il trattamento, la trasformazione, l' "elaborazione" delle informazioni è prerogativa di strumenti più complessi, chiamati appunto Elaboratori.

1.2 Schema generale di un Elaboratore

L'elaborazione delle informazioni si attua mediante la risoluzione da parte dell'Elaboratore di un numero di problemi praticamente illimitato. Si dice: "non c'è problema che un Elaboratore elettronico non sappia risolvere convenientemente", ciò che consente di affermare che l'Elaboratore è "universale".

Non esiste problema senza che vi siano:

1) dei dati di partenza, in base ai quali

2) procedere alla risoluzione e dai quali

3) ricavare i risultati opportuni.

In seguito a ciò ecco la prima suddivisione di un Elaboratore in tre parti fondamentali:

1) Unità di Entrata, detta anche Input;

2) Unità Centrale di Elaborazione;

3) Unità di uscita, dette anche Output.

1.3 Le Unità di Entrata e di Uscita.

Loro funzione è quella di introdurre i dati ed estrarre i risultati di una determinata elaborazione, che si svolge nell'Unità Centrale. Sono costituite dall'insieme delle apparecchiature idonee a trasmettere e a ricevere informazioni, contenute in adeguati supporti capaci di conservarle prima e dopo l'elaborazione.

Queste Unità sono quindi direttamente collegate colla parte Centrale dell'Elaboratore e per questa ragione sono dette Unità in linea (on line), in contrapposizione ad altre Unità che sono fuori dal contatto diretto coll'Elaboratore, del quale non fanno fisicamente parte, perciò chiamate Unità fuori linea (off line).

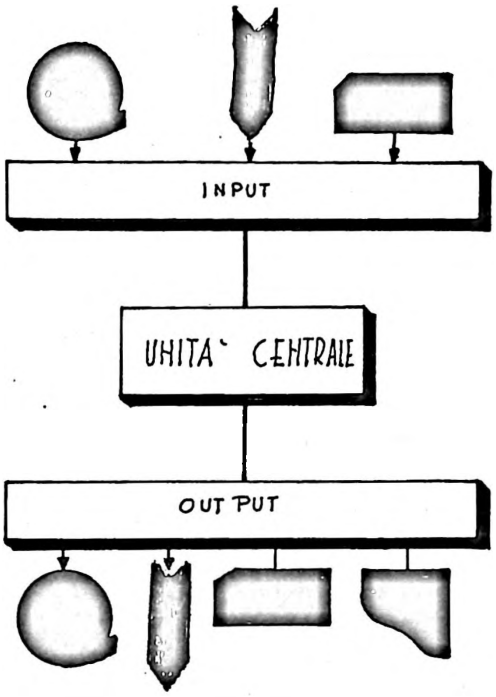
Elenchiamo le principali Unità in linea :

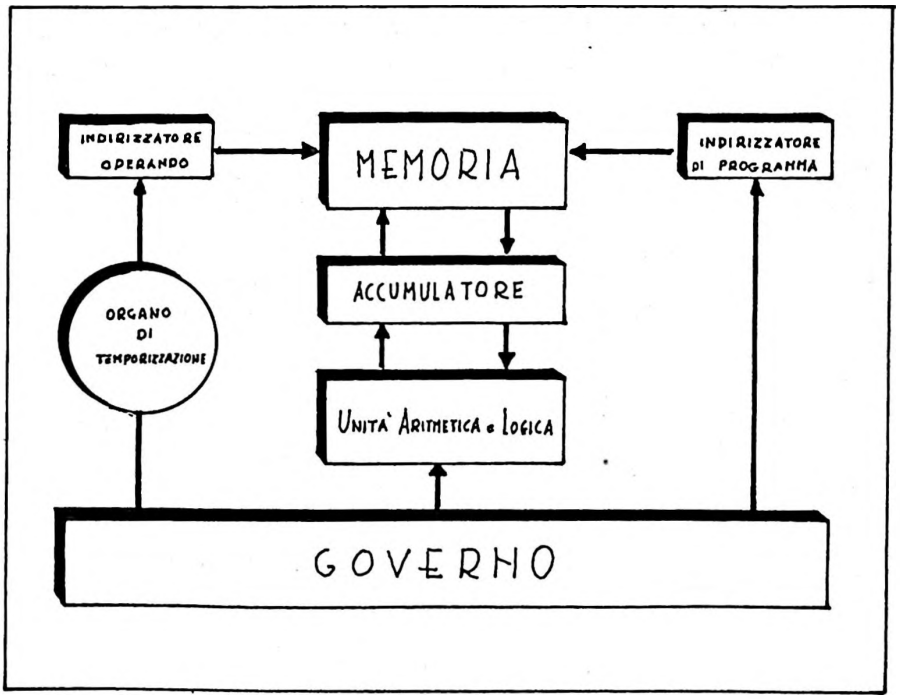
Nome dell'Unità	Input	Output
Lettore di schede	sì	no
Lettore di banda perforata o Fotolettore	sì	no
Perforatore di schede	no	sì
Perforatore di banda	no	sì
Stampanti parallele	no	sì
Unità a nastro magnetico	sì	sì

Dal prospetto si rileva che le Unità a nastro magnetico sono strumenti sia di input che di output, il che significa che la stessa apparecchiatura può, secondo le necessità, leggere il nastro, o registrarlo.

FIGURA 1-1 II

SCHEMA GENERALE
DI
UN ELABORATORE





La struttura tecnica delle varie Unità dipende dal tipo di supporto che esse elaborano, perciò una loro trattazione più specifica sarà data nel capitolo seguente nel quale si parlerà appunto dei supporti di introduzione e di estrazione.

Tuttavia le Unità in linea, proprio perchè collegate con uno stesso Organo Centrale di elaborazione, hanno caratteristiche comuni, che qui esponiamo.

Vi sono degli Elaboratori (tali ad esempio i nostri Elea) che, considerando la lentezza di queste Unità rispetto a quella Centrale di Elaborazione, per non appesantire i tempi di esecuzione del lavoro preferiscono giustamente sganciare il ritmo dell'Unità in linea da quello dell'Unità Centrale. Ciò può avvenire in due modi diversi:

1) si inserisce fra l'Unità in linea e l'Unità Centrale una memoria di transito, detta buffer, di sufficiente capacità, in modo che il caricamento dei dati o l'estrazione dei risultati avvenga alla velocità dell'Unità periferica, mentre lo scaricamento del buffer o il suo riempimento avvengano alla velocità dello organo centrale;

2) si usa per le informazioni di entrata e di uscita, un canale di smistamento particolare distinto da quello attraverso il quale sono elaborate le informazioni dell'Unità Centrale.

In ogni caso l'operazione di introduzione dei dati o di estrazione dei risultati deve essere guidata e controllata da un organo particolare di governo, diverso dal governo dell'Unità centrale.

L' Elaboratore Olivetti Elea 9003 utilizza il secondo procedimento per le Unità a nastro magnetico, il primo per tutte le altre unità.

Qualora il codice dell'Unità in linea sia differente dal codice dell'Unità Centrale (codice macchina dell'Elaboratore) fra le due Unità sarà sempre inserito un codificatore, in modo da effettuare la traduzione dei codici, la quale per talune unità di entrata significherà pure creazione del bit di check.

1.4 L'Unità Centrale

La trasformazione dei dati di partenza in risultati di elaborazione è opera dell'Unità Centrale. Tale funzione è eseguita col concorso dei seguenti organi componenti:

- Memoria
- Unità aritmetica e logica
- Accumulatore
- Indirizzatore di programma
- Indirizzatore di operando
- Organo di temporizzazione
- Governo.

Nei capitoli che seguiranno spiegheremo minutamente ciascuno di essi. La figura 1-2 presenta uno schema illustrative didattico di ciascuno di questi organi; questo schema ci permetterà di esemplificare nel cap. 6° il funzionamento di un Elaboratore ideale.

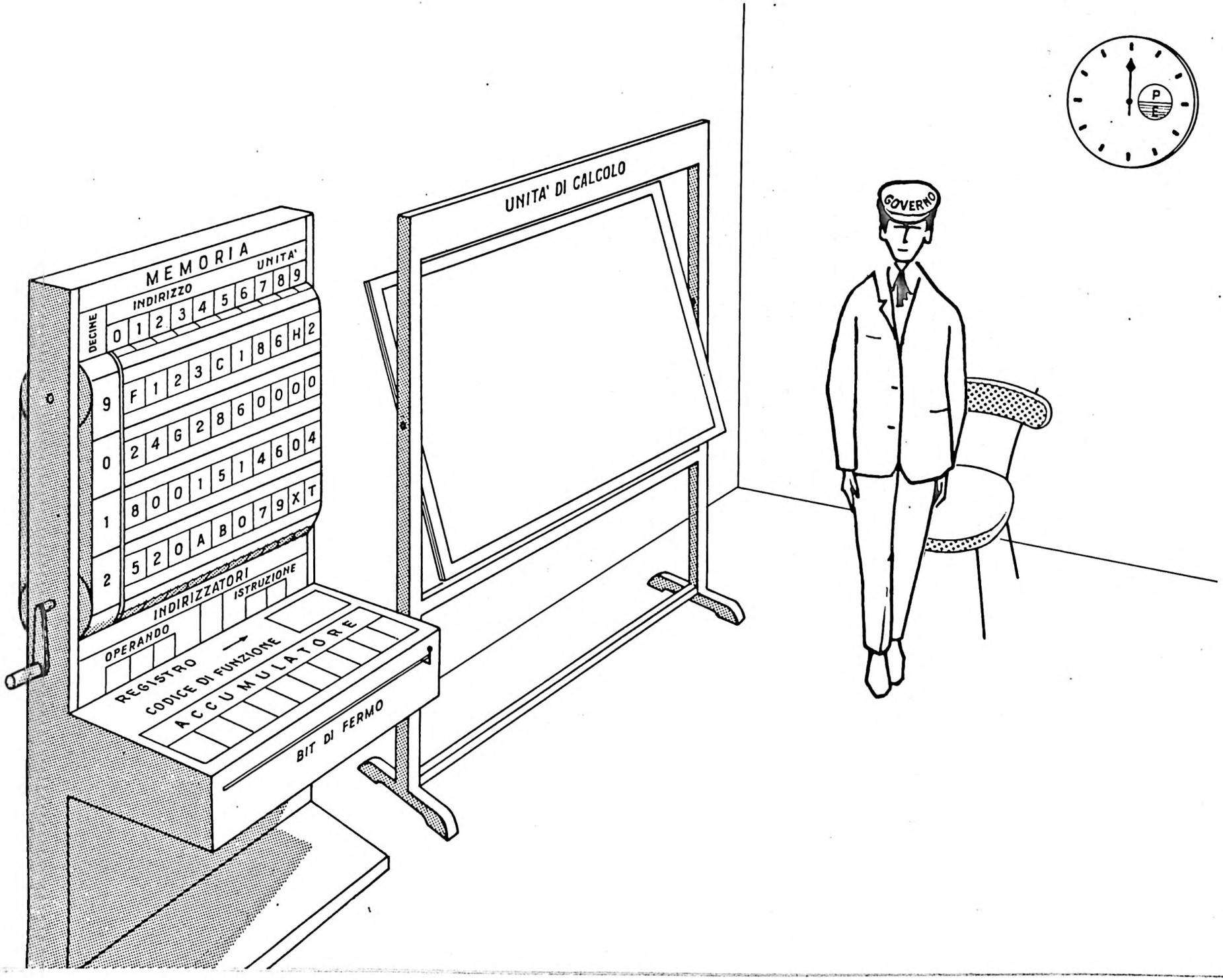

MEMORIA
INDIRIZZO
UNITA'
DECINE
0 1 2 3 4 5 6 7 8 9
9 F 1 2 3 C 1 8 6 H 2
0 2 4 G 2 8 6 0 0 0 0
1 8 0 0 1 5 1 4 6 0 4
2 5 2 0 A B 0 7 9 X T
INDIRIZZATORI
ISTRUZIONE
OPERANDO
REGISTRO
CODICE DI FUNZIONE
ACCUMULATORE
BIT DI FERMO
UNITA' DI CALCOLO
GOVERNO

FIGURA 1-2

E S E R C I Z I

1) Differenza fra Calcolatore e Elaboratore.

2) Differenza fra Unità in linea e fuori linea.

3) Il fotolettore è un'unità di input o di output.

4) Quale è la funzione del buffer?

5) Quali modi conoscete per sganciare i tempi delle Unità in linea da quelli dell'Unità Centrale?

6) Quale è la funzione dell'Unità Centrale?

7) Di quali organi si compone l'Unità Centrale?

II I SUPPORTI

2.1 Definizione

Chiamiamo supporto qualsiasi strumento idoneo a contenere informazioni.

I supporti che interessano gli Elaboratori sono i seguenti:

Supporti d'introduzione :
- Scheda perforata
- Banda perforata
- Nastro magnetico

Supporti d'estrazione :
- Scheda perforata
- Banda perforata
- Modulo stampa
- Nastro magnetico.

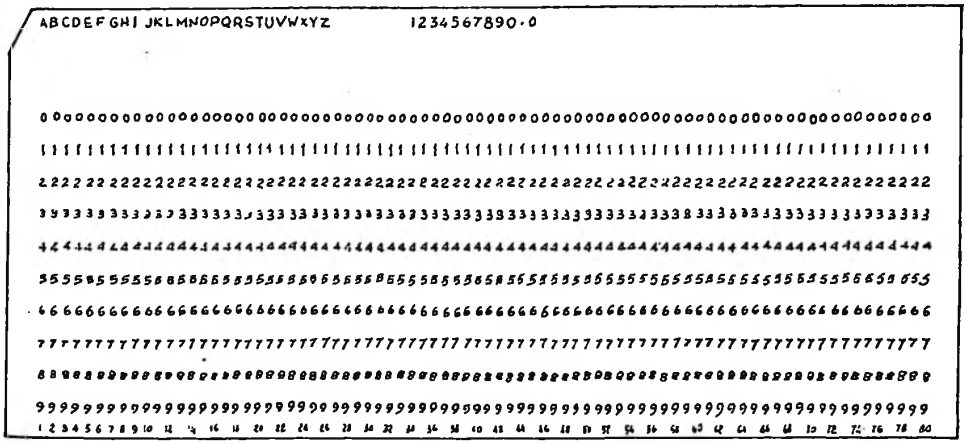
Questi supporti costituiscono il veicolo, mediante il quale i dati di entrata convergono verso l'Unità Centrale ed escono da essa come risultati di elaborazione . Essi sono chiamati anche memorie esterne o memorie lente, vedi cap. 3 - 8.

2.2 Scheda perforata

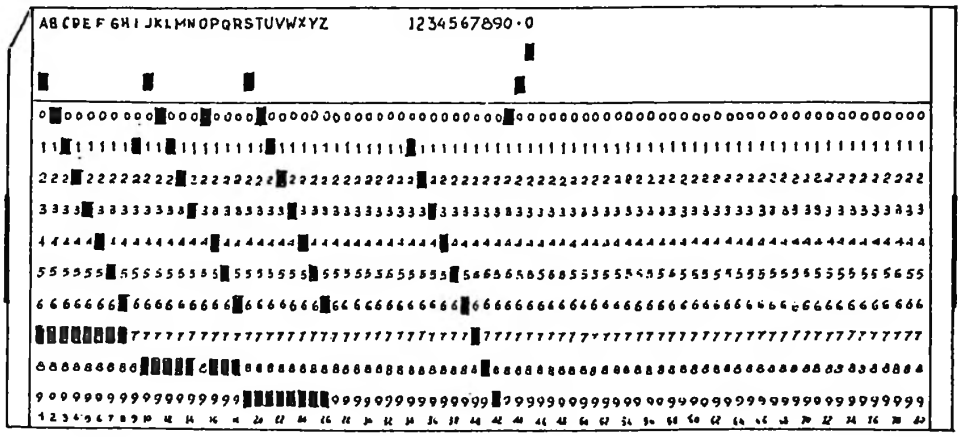
E' costituita da un foglio di cartone speciale, di dimensioni ben definite (187,3 x 82,54 mm), a formato rettangolare, leggermente smussato allo spigolo superiore sinistro. In esso si distinguono 80 colonne verticali, distanti l'una dall'altra 2,21 mm e 12 linee orizzontali distanti l'una dall'altra 6,35 mm. La loro intersezione definisce una piccola superficie rettangolare, nella quale può essere praticato un foro o perforazione. La forma della perforazione è generalmente rettangolare di dimensioni 1,4 x 3,2 mm. Il significato e il valore di questa perforazione consiste nella rappresentazione di un carattere. Possiamo avere un carattere ogni colonna e così tutta la scheda può contenere un massimo di 80 caratteri.

FIGURA 2-1 II

SCHEDA PERFORATA



SCHEDA BIANCA



SCHEDA CON CARATTERI

Come si vede dalla figura 2-1-a), ogni riga ripete lo stesso numero 80 volte ed esse differiscono tutte fra di loro per il diverso numero che ripetono. Le righe numerate però sono solo 10 (da 0 a 9), infatti le righe 11 e 12 non sono scritte e si devono immaginare sopra la riga 0, nella parte bianca. Per questa ragione sono dette anche "sopraelevate".

2.3 I codici scheda

La perforazione può rappresentare due classi diverse di caratteri quelli numerici e quelli alfabetici.

I caratteri numerici sono indicati con una sola perforazione, effettuata in corrispondenza della riga che ne indica il valore.

I caratteri alfabetici sono ottenuti dalla combinazione di due perforazioni nella stessa colonna. Infatti sarebbe stato alquanto oneroso ricorrere ad una scheda che avesse tante righe quanti sono le configurazioni dei caratteri da rappresentare!

La rappresentazione alfabetica è ottenuta combinando due perforazioni, una detta "preparatoria" l'altra detta esecutiva. Tale combinazione è differente secondo il tipo di macchine su cui le schede vengono perforate. Si hanno così i codici Bull e IBM, che illustriamo qui di seguito

	Esecutive								
Codice Bull		11	0	1	2	3	4	5	6
Preparatorie	7	A	B	C	D	E	F	G	H
	8	J	K	L	M	N	O	P	Q
	9	S	T	U	V	W	X	Y	Z

A queste lettere occorre aggiungere 1 = I e 0 (zero) = O. infatti il codice Bull parte dal presupposto che se la parola è numerica, la perforazione sulla riga uno ha valore 1, se la parola

è invece alfabetica la stessa perforazione ha valore I. Parimenti dicasi per la perforazione sulla riga zero. La perforazione 12 non è utilizzata per caratteri, ma per comandare determinati organi e per ottenere alcune segnalazioni.

		Esecutive								
Codice IBM		1	2	3	4	5	6	7	8	9
Preparatorie	12	A	B	C	D	E	F	G	H	I
	11	J	K	L	M	N	O	P	Q	R
	0		S	T	U	V	W	X	Y	Z

Come si vede questo codice usa un carattere esecutivo in più, che gli consente di rappresentare altri tre caratteri, di cui ne utilizza solo due che servono per la I e la O, che il codice Bull omette. La combinazione 0,1 non rappresenta nessun carattere.

2.4 Perforazione della scheda

La perforazione delle schede avviene mediante apposite macchine, che comandano l'abbassamento di un punzone (o di due punzoni) in corrispondenza della riga (o delle due righe) colla colonna in cui si desidera registrare il carattere. Ovviamente per i caratteri numerici si abbasserà un solo punzone, per quelli alfabetici se ne abbasseranno due.

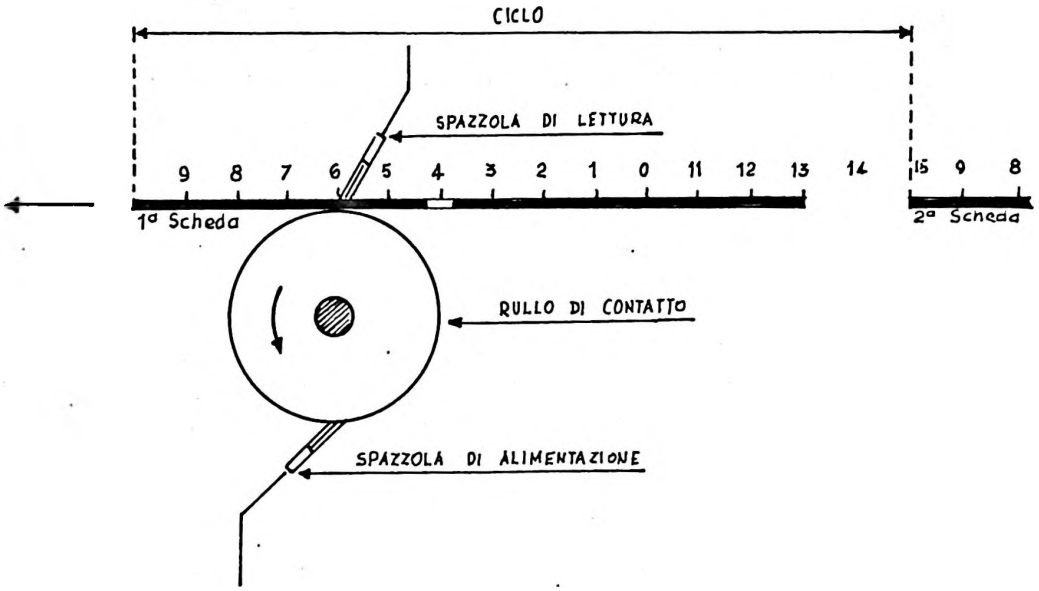
La figura 2.1.b mostra perforati in codice Bull tutti i caratteri alfabetici e numerici.

L'operazione di perforazione può avvenire sia a mano che automaticamente con macchine chiamate Perforatori.

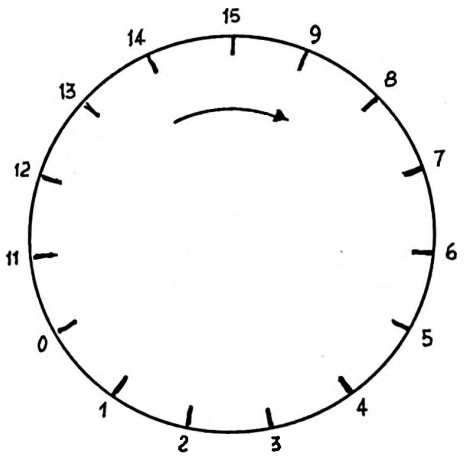
Vi sono perforatori collegabili coll'Elaboratore i quali automaticamente registrano su scheda informazioni provenienti dall' Unità centrale.

FIGURA 2-2 II

SCHEMA DI LETTURA DI UNA SCHEDA



PUNTI DEL CILINDRO DI ROTAZIONE.



Con queste macchine (contrariamente a quello che succede per la perforazione manuale) la perforazione avviene normalmente in parallelo ad una velocità che varia, secondo il tipo di unità da 80 a 300 schede al minuto. I perforatori sono dotati di dispositivi di controllo atti a segnalare eventuali errori di perforazione.

2.5 Lettura della scheda

La scheda viene letta in movimento. Trascinata da appositi rulli, essa passa tra un cilindro rotante e una spazzola di lettura. La spazzola è montata in appositi alloggiamenti in modo che i suoi spazzolini metallici possano toccare la superficie del cilindro, denominato cilindro di alimentazione.

La lettura avviene in questo modo. Appena la scheda arriva sotto gli spazzolini, il cilindro di alimentazione riceve degli impulsi elettrici. Poichè la scheda è fatta di materiale isolante, impedisce che gli impulsi vengano raccolti dagli spazzolini. Ma appena vi è una perforazione lo spazzolino va a toccare il cilindro d'alimentazione e di conseguenza riceve un impulso di corrente.

Tuttavia se l'impulso è sufficiente ad individuare una perforazione, deve essere integrato con qualcos'altro per stabilire il valore della perforazione letta; il quale è in diretta relazione con la posizione del foro nella colonna.

A questo proposito dobbiamo introdurre il concetto di ciclo.

Per ciclo s'intende il tempo che impiega il cilindro (solidale con un albero rotante, detto albero generale) a compiere un giro completo. Il ciclo viene diviso in più parti, chiamate punti, che normalmente sono 15 o 16, ma possono arrivare fino a 18. A dodici di questi punti vengono fatte corrispondere le 12 posizioni di perforazione delle 80 colonne della scheda, per cui, a seconda del punto in cui giunge l'impulso alla macchina, essa è in grado di interpretare il valore della perforazione. La lettura risulta dunque da una combinazione spazio-tempo, rispettivamente data dalla perforazione e dal punto. I rimanenti punti chiamati punti morti, sono intervalli di lettura fra una scheda e l'altra durante i quali l'Unità meccanografica compie l'elaborazione della scheda letta.

Poichè 80 sono gli spazzolini, è chiaro che le colonne vengono esplorate tutte insieme, si ha così la lettura dei caratteri della scheda in parallelo.

Il lettore di schede è un'Unità collegata in linea coll'Elaboratore, la cui funzione è quella di trasferire il contenuto della scheda perforata nella Unità centrale del sistema, esattamente nella memoria principale.

Essi leggono le schede a velocità variabili, secondo il tipo di unità, da 200 a 700 schede al minuto primo e sono dotati di sistemi di controllo per rilevare eventuali errori di lettura.

2.6 La banda perforata

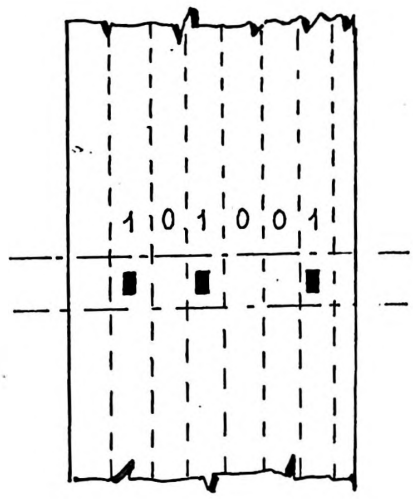
La banda perforata è un lungo nastro di carta bianca, avvolgibile intorno ad una bobina. La sua larghezza è di mm 20,5, mentre la sua lunghezza varia corrispondentemente alla capacità della bobina di svolgimento, che tuttavia di norma è di 150 metri.

La banda perforata, come le corsie di una grande arteria, è suddivisa in 6 piste ideali parallele, più due bordi laterali, figura 2-3.

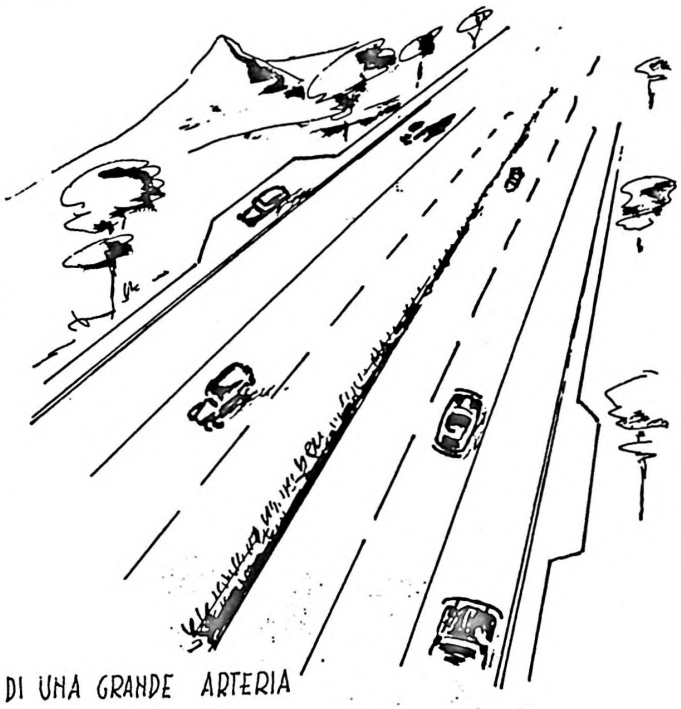
Se intersechiamo la banda perforata con due segmenti paralleli distanti 3 mm fra di loro, l'intersezione delle piste con i segmenti dà luogo a sei superfici rettangolari, ognuna delle quali costituisce un bit. Esso assume valore 0, se la superficie rimane bianca, 1 se la superficie è perforata mediante un foro di mm. 2,05 x 1,65. La combinazione simultanea dei sei bit rappresenta un carattere e 64 sono le configurazioni diverse di caratteri che si possono perforare.

La perforazione dei sei canali rispetto alla larghezza del nastro perforato è leggermente spostata verso destra, pertanto il margine del nastro non perforato è maggiore a sinistra. Questo serve a distinguere il lato giusto della banda rispetto al dispositivo di lettura.

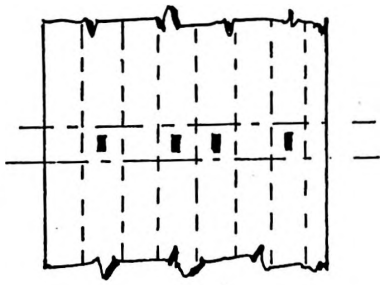
FIGURA 2-3 II

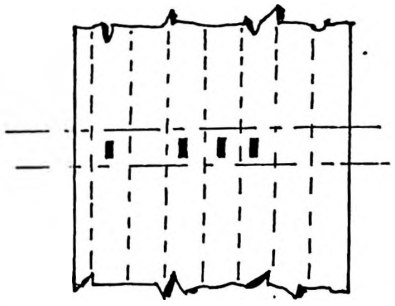


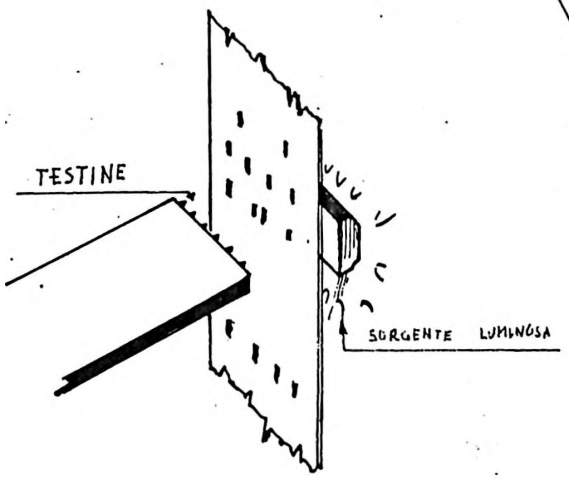
LA BANDA

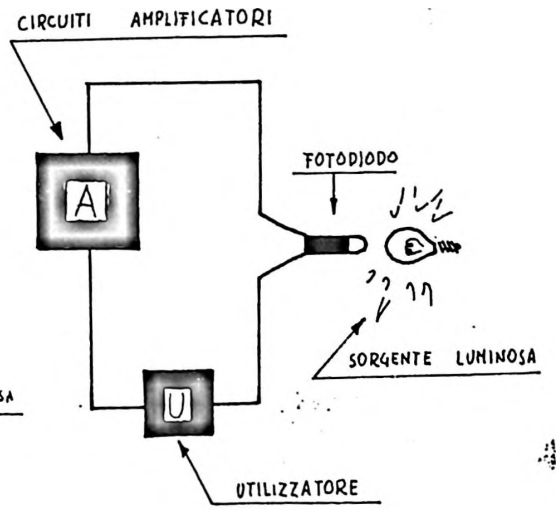


LE CORSIE DI UNA GRANDE ARTERIA









2.7 Registrazione su banda

La registrazione su banda è dovuta a perforazione mediante un gruppo di sei punzoni che vengono posizionati al momento dell'abbassamento del tasto corrispondente al carattere che si desidera trascrivere e che quindi perforano la banda nei bit 1.

La perforazione avviene dunque in serie,cioè carattere per carattere. E l'apparecchio più comune è una macchina da scrivere e da calcolo che ha appunto annesso il blocco perforatore. Esso presenta il vantaggio di una perforazione simultanea alla compilazione del documento, il costo della quale risulta praticamente nullo.

2.8 Lettura su banda

Per la lettura della banda perforata si utilizza un gruppo di sei fotodiodi al germanio, uno per ciascuna pista della banda.

La fotocellula al germanio, detta comunemente testina, ha la proprietà di emettere elettroni, se investita da un fascio luminoso. Se fra le testine e una sorgente luminosa si intercala la banda perforata in movimento, è chiaro che in corrispondenza delle perforazioni si produrrà passaggio di elettroni della fotocellula, mentre in corrispondenza delle non perforazioni la rispettiva fotocellula non verrà impressionata.

Anche la lettura come la perforazione avviene in serie, in quanto la banda si svolge in continuità da una bobina superiore, passa attraverso le testine, per riavvolgersi in una bobina inferiore, che inizialmente sarà vuota.

L'Unità, capace di leggere la banda e di introdurne i caratteri nella parte Centrale dell'Elaboratore, è detta Lettore di banda perforata o Fotolettore. La velocità di queste apparecchiature varia dai 600 agli 800 caratteri al secondo, in rapporto alle caratteristiche tecniche dell'Unità impiegata.

2.9 Il nastro magnetico

Il nastro magnetico è costituito da un supporto plastico della larghezza di $\frac{1}{2}$ pollice (cioè mm 12,7), rivestito da uno strato sottilissimo ed uniforme di ossido di ferro, avvolto in una bobina, che può raggiungere in lunghezza la dimensione massima di 1100 metri .

Anche il nastro magnetico, similmente alla banda perforata, è suddiviso in un certo numero di piste longitudinali. Sezionando in modo trasversale il nastro magnetico le intersezioni con ognuna delle piste individuano un bit.

Tutti i bit della intersezione concorrono a determinare un carattere.

Il numero delle piste presenti su un determinato nastro è funzione delle configurazioni di caratteri che si intendono rappresentare .

Inoltre si ha una pista per bit di controllo e una pista per bit di temporizzazione. Esamineremo più tardi la funzione del bit di temporizzazione.

I bit su nastro magnetico si ottengono mediante aree magnetizzate. Si assume come valore 1 dello stato di magnetizzazione l'orientamento in un senso delle particelle e come valore 0 l'orientamento in senso opposto (nord e sud).

A questo punto è necessario dare al fenomeno una spiegazione più esauriente.

2.10 L'induzione magnetica e il ciclo di isteresi

Si supponga di avere un nucleo di ferrite (ossido di ferro) e di avvolgergli attorno un certo numero di spire.

Facciamo passare lungo le spire una corrente di intensità i e studiamo la distribuzione del flusso magnetico ϕ nel nucleo, funzione della variabile i. Nel grafico (figura 2.4-a) metteremo i in ascissa e ϕ in ordinata.

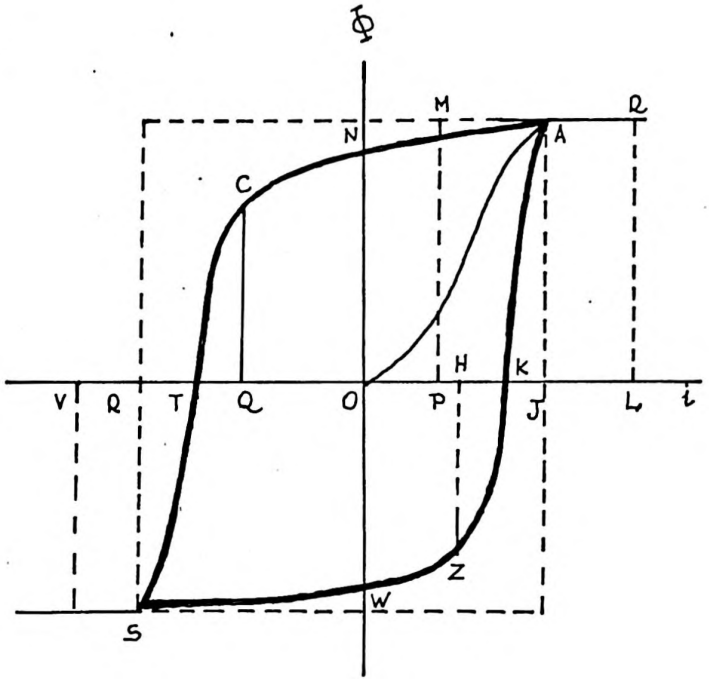
Supponiamo che all'inizio il materiale non sia affatto magnetizzato e facciamo crescere l'intensità di corrente, fino ad assumere il

FIGURA 2 - 4

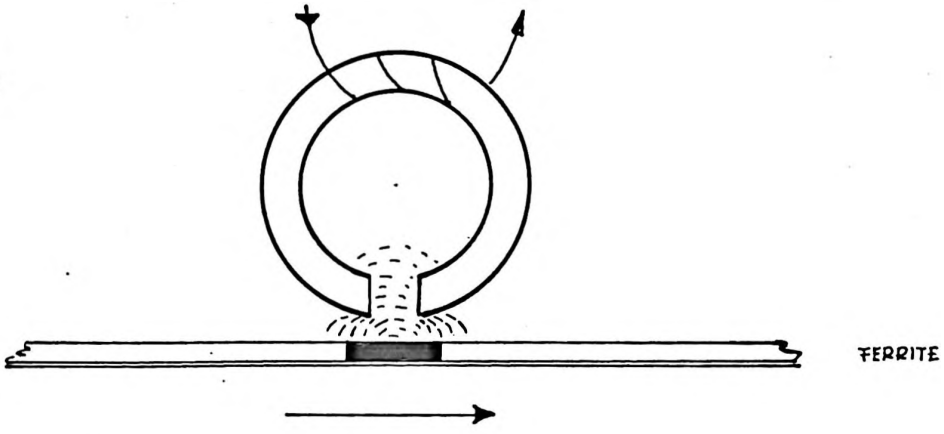
II

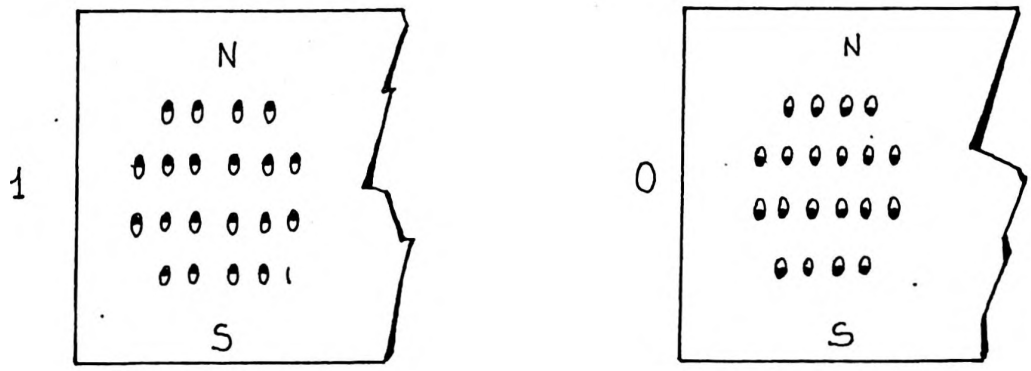
CICLO DI ISTERESI RETTANGOLARE

a)



b)





AREOLA MAGNETIZZATA

AREOLA SMAGNETIZZATA

valore J. Si osserva che il flusso magnetico che attraversa il corpo disegna la curva OA. Se aumenta l'intensità da J a L, il flusso di magnetizzazione B si mantiene pressochè costante e uguale ad A.

Supponiamo adesso che la corrente decresca da J a P, il flusso magnetico presente nel corpo si stabilisca assai vicino al valore A precedente, cioè ad M. Se addirittura l'intensità passa a O, ϕ si arresta al valore N.

Facciamo crescere l'intensità negativamente e notiamo che fino al valore di ϕ decresce leggermente; però precipita dal valore C al valore T per il brevissimo spostamento QT della intensità, quindi inverte il senso e nuovamente, al decrescimento RT di i, discende per valori negativi fino ad e S, dal cui valore non si discosta più.

Analogamente, se da R i passa a O e ad H, il flusso descrive il tratto SWZ, dove il valore di ϕ si discosta di poco da S. Ma anche qui, se i aumenta da H a K e da K a J, si ha precipitazione di flusso magnetico al valore opposto A.

I punti A ed S sono detti punti di saturazione magnetica, infatti il valore di ϕ rimane fisso e costante ad una variazione intensiva di i. I punti N e W si chiamano punti di "induzione residua". Essi infatti individuano la quantità di energia magnetica che rimane presente nel corpo al cessare del fenomeno i che l'ha generata.

Il flusso magnetico generato dalla sfera ha una direzione che è opposta a quella della corrente che lo ha prodotto. Esso provoca l'orientamento delle particelle (figura 2.4-b) nel senso nord, quando l'intensità è positiva e uguale a J e il flusso è vicino al punto A. L'orientamento avviene invece in senso sud, quando la intensità passa a T e il flusso s'avvicina al punto S. Venuta meno la corrente inducente, sul corpo rimane il flusso residuo. E' questa una proprietà fondamentale dei materiali ferromagnetici .

2.11 Le correnti indotte

Una variazione di flusso magnetico attraverso un conduttore genera una corrente elettrica in quest'ultimo, detta corrente indotta.

Siamo in presenza del fenomeno inverso a quello precedentemente visto dell'induzione magnetica. Per il quale valgono le stesse regole del ciclo d'isteresi, avendo cura di invertire le ordinate colle ascisse. Parimenti il senso della corrente generato nel conduttore si oppone a quello delle linee di flusso magnetico che lo hanno generato.

2.12 Registrazione e lettura su nastro magnetico

La registrazione su nastro magnetico avviene inviando impulsi di corrente, che, magnetizzando un piccolo indotto a ferro di cavallo (detto testina), provoca l'induzione per influenza della corrispondente superficie (areola) di nastro. La quale, essendo in movimento, avrà dei punti in cui il flusso è più intenso e altri in cui il flusso è minore, (figura 2.5-a). Ogni testina registra un bit.

La variazione di flusso di fronte ad una testina elettromagnetica, provoca la lettura del bit di nastro, generando una corrente indotta fra le spire che avvolgono la testina.

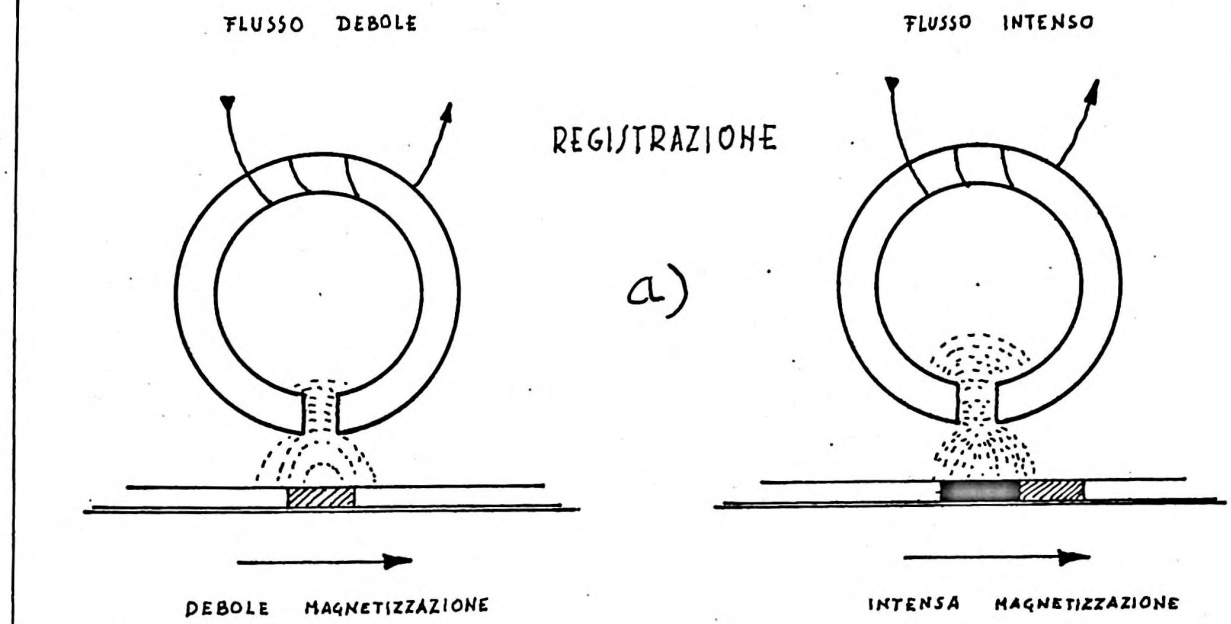
La figura 2.5-B, nelle tre rappresentazioni successive, mostra come il nastro col suo movimento provochi la variazione di flusso dentro la testina di lettura, condizione perchè nella spira si abbia un impulso di corrente.

Il senso di movimento di tale impulso distinguerà il valore 1 o 0 del bit letto.

Su nastro magnetico la registrazione e la lettura del singolo carattere avviene in parallelo, si hanno pertanto gruppi di testine di registrazione e gruppi di testine di lettura con tanti elementi quanti sono i bit.

La registrazione e la lettura dei vari caratteri avviene invece in serie, mediante scorrimento del nastro sotto le testine.

FIGURA 2-5


FLUSSO DEBOLE
FLUSSO INTENSO
REGISTRAZIONE
a)
DEBOLE MAGNETIZZAZIONE
INTENSA MAGNETIZZAZIONE

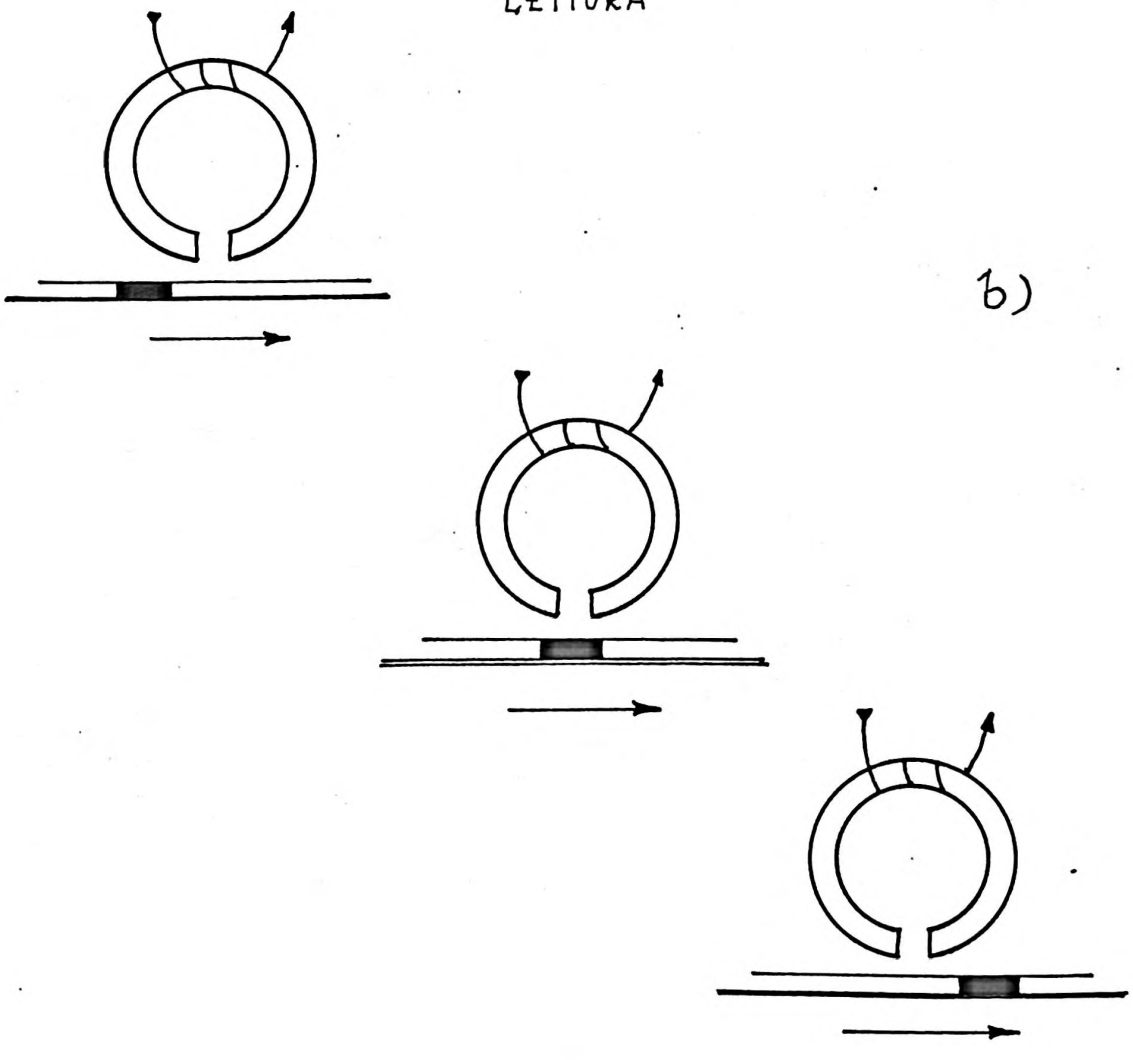
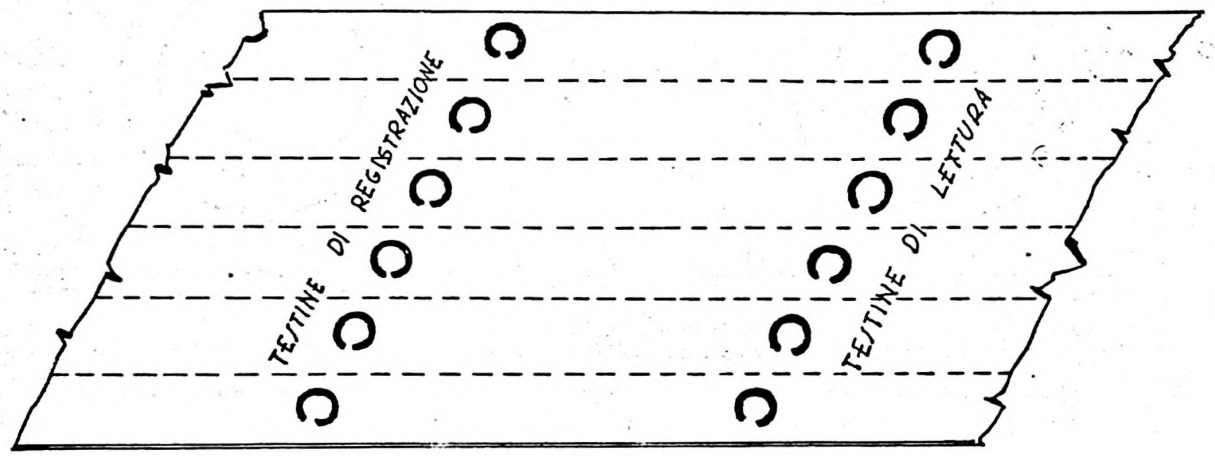

LETTURA
b)

FIGURA 2-6 II

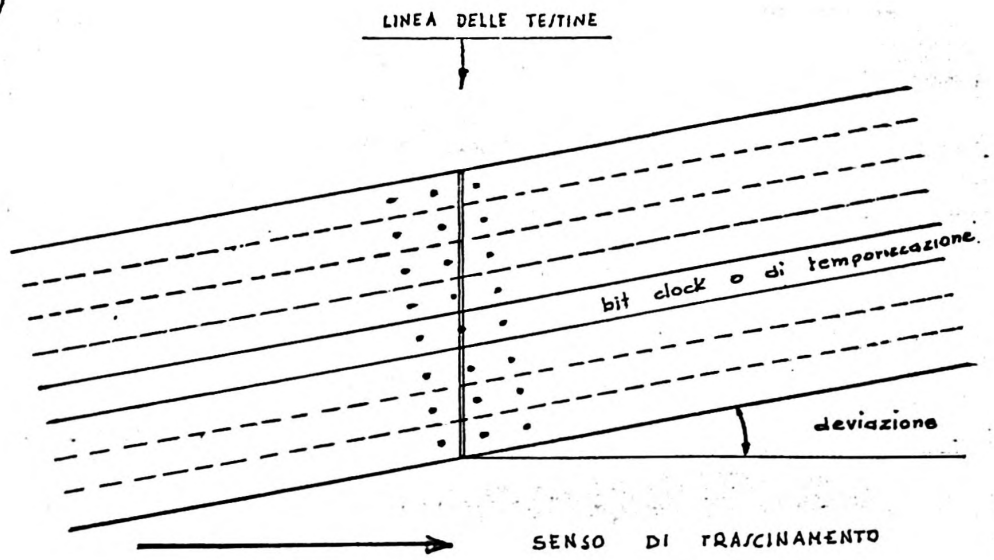
GRUPPI DI TESTINE

A)



IL BIT DI TEMPORIZZAZIONE

B)



Il numero di caratteri, che è possibile registrare (e di conseguenza leggere) sull'unità di lunghezza del nastro magnetico, definisce la densità o impaccamento del nastro.

Gli strumenti attraverso i quali i caratteri passano nella parte centrale dell'Elaboratore, oppure dall'Elaboratore vanno a registrarsi su nastro sono chiamate Unità a Nastro Magnetico. Come si è detto nel capitolo precedente la stessa Unità a nastro è in grado sia di leggere che di registrare. Il fenomeno avviene alla medesima velocità, la quale, secondo il tipo di Unità usate, varia da 22.500 a 45.000 e persino 90.000 caratteri al secondo!

2.13 Il bit di temporizzazione

Abbiamo visto che, perchè sia possibile la lettura e la registrazione su nastro magnetico, occorre che questo scorra sotto la testina mediante trascinamento. La velocità di trascinamento è molto elevata: può raggiungere i m 3,8 al secondo e persino m 7,5! Non solo, ma poichè, come studieremo, il nastro verrà letto e registrato a pezzetti (blocchi) della lunghezza di 30/40/50 cm. di media, risulta evidente che il nastro può essere soggetto a vibrazioni durante il trascinamento. Per questa ragione può succedere che il nastro arrivi sotto le testine coi bit in posizione diagonale, per cui non vengono letti (o registrati) tutti contemporaneamente, ma con un leggero sfasamento.

Evidentemente lo sfasamento non dovrà superare certi limiti, nel senso che prima di terminare la lettura dell'ultimo bit di un carattere non accadrà che si sia letto un bit del carattere successivo. Lo sfasamento è dunque contenuto entro margini di sicurezza. -

Ciò premesso in figura 2.6-b il bit H sarà stato letto per primo successivamente saranno stati letti rispettivamente G F D C B e infine il bit A. E' necessario in questo caso "temporizzare l'operazione" cioè fissare un limite di tempo a questi ritardi di sfasamento, dopo il quale si è senz'altro sicuri di aver letto tutti i bit. Questo tempo è fisso ed è conteggiato a partire dalla lettura di un bit centrale, il quale, data la sua posizione, subisce vibrazioni molto minori (direi trascurabili) ed è detto bit di clock

o di temporizzazione. Analoghi considerazioni valgono per la registrazione, dove però il bit clock viene posto su nastro dalla testina centrale.

2.14 Organizzazione dei dati su nastro magnetico

I dati su nastro magnetico richiedono una particolare organizzazione, dovuta a due ragioni fondamentali:

1) Il rapporto fra nastro magnetico (supporto valevole come memoria d'archivio) e la memoria principale (organo attraverso il quale devono obbligatoriamente passare tutti i dati destinati all'elaborazione) è molto elevato. Si pensi ad esempio ai 13.000.000 di caratteri che contiene una bobina di nastro in confronto dei 10/40.000 della memoria di lavoro. Per questa ragione il nastro deve essere suddiviso a gruppi d'informazioni detti blocchi.

2) Il nastro viene letto un carattere dopo l'altro e la posizione successiva di lettura è legata alla posizione immediatamente precedente, per cui le informazioni debbono essere distribuite in una sequenza ordinata.

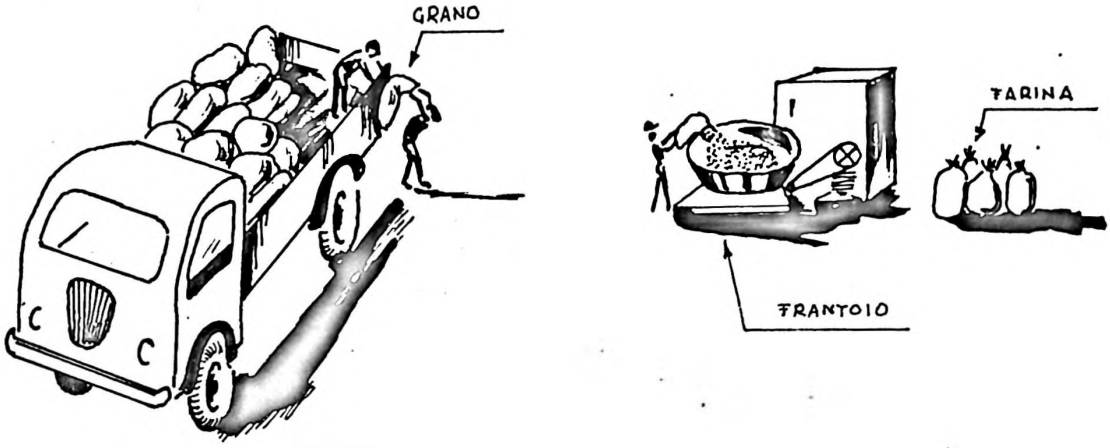
Spieghiamo questi due punti.

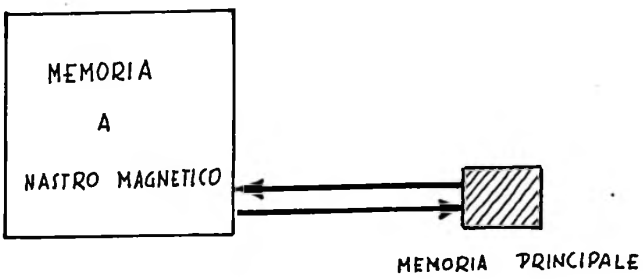
E' chiaro che i dati presenti su nastro non possono essere caricati completamente in memoria di lavoro in quanto la capacità di questa è molto limitata. Occorre dunque dosarli in modo da farli entrare e uscire in numero limitato, in rapporto alla disponibilità di spazio nella memoria principale.

Si supponga un lavatoio reggimentale, al quale debbono accedere interi battaglioni di reclute per il bagno. Se il locale ha la capacità di 30 docce, il capitano d'ispezione darà disposizioni, perchè le reclute entrino a gruppi di 30 per volta e in pari numero per non generare confusione provvedano ad uscire (figura 2.7-B). Lo stesso accade per i nastri magnetici, sui quali i dati si dispongono a gruppi chiamati blocchi. La lunghezza in caratteri dei blocchi è condizionata dal concorso di 3 fattori :

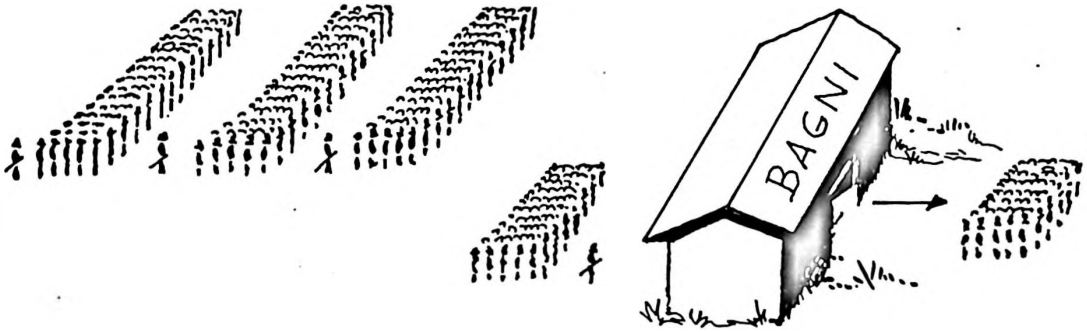
FIGURA 2 - 7

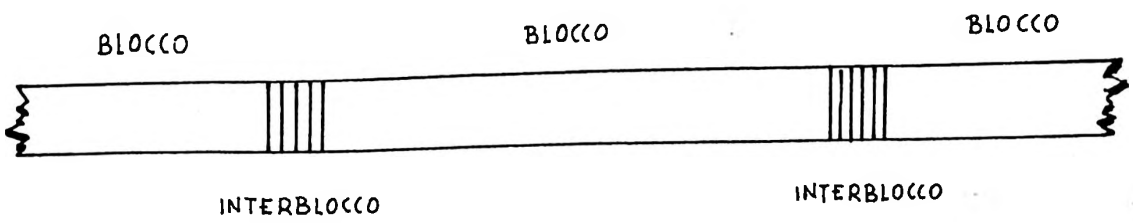
a)


GRANO
FARINA
FRANTOIO


MEMORIA
A
NASTRO MAGNETICO
MEMORIA PRINCIPALE

b)


BAGNI


BLOCCO
BLOCCO
BLOCCO
INTERBLOCCO
INTERBLOCCO

1) Capacità della memoria principale,

2) Dimensioni del programma,

3) Numero di altre Unità a nastro che intervengono durante l'elaborazione o le elaborazioni, a cui quel particolare nastro partecipa.

Ovviamente, la lunghezza dei blocchi è in proporzione diretta col caso 1 e in proporzione inversa coi casi 2 e 3.

Per quanto riguarda la sequenza ordinata dei dati, essa è in rapporto al fatto che il nastro è una memoria sequenziale (vedi cap. 3 . 2), la quale per essere convenientemente utilizzata deve presentare le informazioni ordinate secondo un certo codice.

Esempio il nastro "matricola salariati", che contiene tutti i dati anagrafici relativi agli operai della ditta x, risulterà ordinato per numero di matricola. Leggendo il nastro incontreremo prima le informazioni dell'operaio 00100 quindi quelle dell' operaio 00500 ecc. .

2.15 Gli interblocchi

Il plotoncino di reclute che si recava a fare la doccia entrava e usciva a scaglioni; ovviamente, poichè restavano per circa quindici minuti nel lavatoio, il successivo scaglione era costretto a sostare in attesa.

Lo stesso avviene per i blocchi di nastro in entrata e uscita. Entra in blocco in memoria principale, rimane un certo tempo per la elaborazione, quindi esce elaborato su un altro nastro.

Intanto però il blocco successivo è costretto ad aspettare. Per questa ragione il nastro risulterà in movimento durante la lettura del blocco, ma mentre questo è in memoria principale per l'elaborazione, il nastro starà fermo in attesa del momento di lettura del blocco seguente.

Lo stesso accade in registrazione: esso risulterà in movimento , finchè registra; successivamente starà fermo per attendere l'elaborazione del blocco successivo che dovrà poi registrare.

Il movimento del nastro si presenta dunque con una successione alternata e ripetuta di pause, per cui non è errato dire che il nastro procede a scatti.

Questo modo di procedere implica delle fasi di accelerazione o di start e delle fasi di arresto o di stop. Infatti per iniziare la lettura del blocco è necessario che il nastro scorra sotto le testine a velocità di regime, la quale ovviamente, trattandosi di trascinamento meccanico, non è raggiunta se non dopo un certo periodo di tempo, chiamato tempo di accelerazione o di start. Durante l'accelerazione il nastro non può assolutamente leggere, perciò lo spazio corrispondente è privo di caratteri, perchè, se vi fossero, andrebbero senz'altro perduti.

Terminata la lettura dell'ultimo carattere del blocco, il nastro si deve fermare. Succede tuttavia che per inerzia procede in avanti per un certo tempo, detto tempo di decelerazione o di stop, durante il quale il nastro percorre un tratto in cui non vi debbono essere caratteri da leggere.

Lo stesso accade in registrazione.

Lo spazio di nastro intercalato fra ogni blocco per coprire i tempi di start-stop è detto interblocco o gap. Esso rappresenta una parte di nastro completamente inutilizzabile, per cui la misura di sfruttamento della capacità di un nastro è data dal numero dei blocchi e quello soltanto. Infatti dalla sua capacità teorica è necessario togliere il numero d'interblocchi, il quale è tanto maggiore quanto minore è la dimensione media di ciascun blocco.

2.16 Caratteri particolari di nastro

Su nastro magnetico vi sono dei caratteri speciali con funzioni di servizio.

Esempio: nell'Elea 9003 il carattere " ¢ " e " ? " costituiscono i caratteri di inizio e di fine blocco. La lettura di "?" provo-

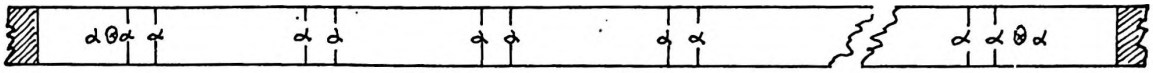
FIGURA 2 - 8

II

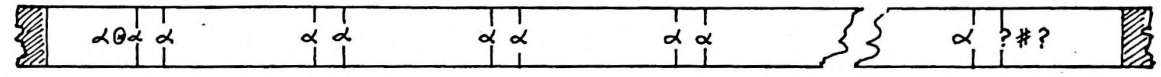
ELEA 9003

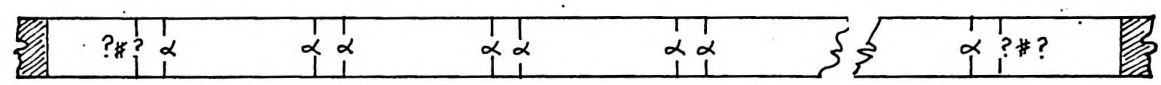
DATI SU NASTRO MAGNETICO

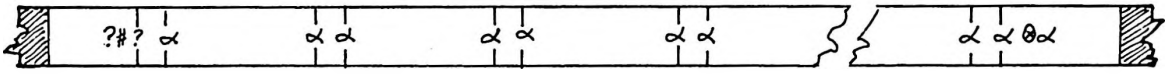
FLUSSO DI UNA SOLA BOBINA



FLUSSO DI 3 BOBINE

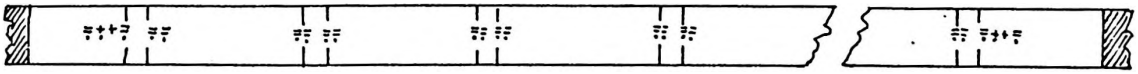




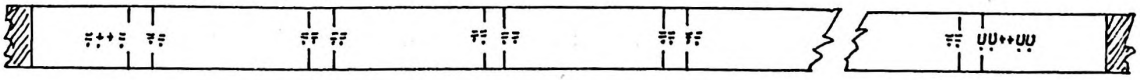


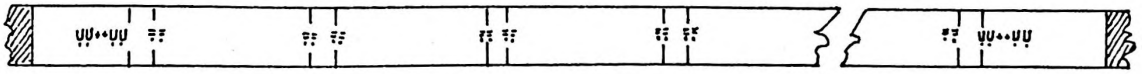
ELEA 6001

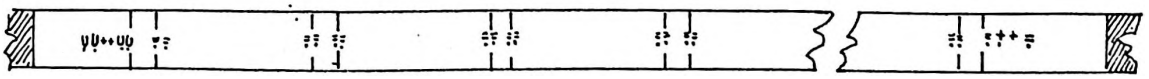
FLUSSO DI UNA SOLA BOBINA



FLUSSO DI 3 BOBINE







ca inoltre l'emissione di un segnale di macchina e lo stesso la lettura di ⊗ (carattere che si legge moltiplicato - per).

I programmatori possono dunque utilizzare questi segnali, attri - buendo ad essi significati particolari. I nostri programmatori nell'Elea 9003 hanno dato al "?" il valore di fine sequenza e a ⊗ il valore di fine flusso.

Supponiamo di avere un flusso di informazioni, il cui contenuto è distribuito su tre bobine di nastro magnetico. I dati, sulle rispettive bobine risulteranno nell'Elea 9003 così organizzati, come si vede nella figura 2.8-a.

I segnali di fine sequenza "?" che coincidono col fine bobina sono disposti in modo da funzionare sia se il nastro è letto in senso avanti, (freccia A) o in senso indietro (freccia B). Ovviamente, se il nastro è letto in senso avanti la successione è bobina 1, bobina 2, bobina 3; se è letto in senso indietro è bobina 3, bobina 2, bobina 1.

Il blocco ∝—⊗-∝ è il blocco di inizio e di fine; molto spesso contiene dati statistici o di riferimento.

Analogo discorso può ripetersi per l'organizzazione dei nastri sull'Elea 6001, dove i caratteri particolari sono diversi, ma il significato non cambia.

2.17 Il modulo stampa

La descrizione di questo supporto è molto semplice, in quanto consiste in un foglio di carta che può essere completamente bianco oppure può presentare fincature e descrizioni varie prestampate.

Su questo foglio vengono stampati documenti e prospetti di valore contabile e giuridico di diverso tipo e formato, esempi: ruolini paga, tratte, fatture, prospetto di conto corrente, buoni di magazzino ecc.

La stampante collegata in linea coll'Elaboratore è l'unità mediante la quale i dati in uscita vengono scritti sul modulo, che per

comodità di caduta è disposto a soffietto.

I caratteri sono stampati tutti in lettere maiuscole, riga per riga, ragione per cui le Unità di stampa si chiamano anche stampanti parallele. Infatti il blocco di stampa è organizzato in modo tale, che, nell'ambito della riga, prima scrive tutte le lettere di uno stesso tipo, ad esempio le A, poi tutte quelle di un nuovo tipo, ad esempio le B e così via.

Una riga comprende al massimo 100-120 caratteri e la velocità di stampa può variare da 150, 300, fino a 600 righe alfanumeriche al minuto primo, secondo il tipo di stampante collegata.

2.18 Alcune considerazioni pratiche

Dal punto di vista dell'utilizzazione pratica i supporti esaminati presentano alcune caratteristiche importanti. E' necessario infatti sottolineare il grande vantaggio conseguito dall'impiego dei nastri magnetici, i quali oltre al pregio di una velocità enormemente più elevata e di una maggiore sicurezza di elaborazione hanno la caratteristica non meno importante dell'economicità. Infatti mentre la scheda e la banda perforata non possono contenere in tempi diversi informazioni diverse, il nastro magnetico può venire registrato un numero praticamente illimitato di volte. Infatti la registrazione di nuovi caratteri su nastro cancella automaticamente i caratteri precedenti, senza danneggiare il supporto, lo stesso come avviene per il registratore magnetico. La scheda e la banda perforata invece, una volta che le informazioni in esse conservate non servono più all'elaborazione, devono essere buttate via, senza possibilità di utilizzazione alcuna.

Per il nastro magnetico questo non succede, con grande vantaggio dei Centri che lo impiegano.

A ciò si aggiunga le modestissime dimensioni di ingombro che esso presenta, particolarmente nei confronti della scheda. Infatti una bobina di nastro occupa meno di un volume dell'Enciclopedia Treccani. mentre per archiviare lo stesso numero di caratteri perforati su schede occorrerebbero parecchi classificatori ad armadio.

Purtroppo il nastro magnetico contrariamente alla scheda e alla banda ha la remora di non poter essere "caricato" (cioè registrato) manualmente, perciò la formazione dei primi nastri dovrà essere effettuata o direttamente da elaboratore convertendo le informazioni da scheda o da banda perforata, oppure al di fuori del calcolatore con apparecchi speciali chiamati convertitori.

Si è visto al punto 2.7 che la banda presenta il vantaggio di un costo di perforazione manuale praticamente nullo.

E S E R C I Z I

1) Cos'è una scheda?

2) Quante colonne ha una scheda?

3) Quanti caratteri si possono registrare come massimo su una scheda?

4) La lettura della scheda avviene in serie o in parallelo?

5) Nella banda perforata i caratteri sono il risultato di una combinazione binaria?

6) Quale vantaggio economico offre la banda perforata?

7) Cos'è una fotocellula?

8) Perchè si usano materiali magnetici a ciclo d'isteresi quasi rettangolare?

9) Funzione del bit di temporizzazione.

10) Per quale motivo si debbono suddividere i nastri in blocchi?

11) Cos'è un interblocco?

12) Cosa sono i tempi di start e di stop?

13) Cos'è l'impaccamento?

14) Quale è la velocità con cui il nastro magnetico viene letto o registrato?

15) Quali fattori condizionano la lunghezza di un blocco? Perchè?

16) Che differenza c'è fra capacità teorica e capacità effettiva di una bobina di nastro.

17) Perchè per essere letto o registrato il nastro magnetico deve scorrere?

18) Il nastro magnetico può essere registrato manualmente?

19) Come si provvede a caricare i primi nastri?

20) Quali vantaggi presenta il nastro magnetico rispetto agli altri supporti?

III LE MEMORIE

3.1 Generalità

Costituisce "memoria" qualsiasi organo in grado di conservare dati e informazioni, cioè qualsiasi organo in cui è possibile "portare" dei dati e all'occasione "prelevarli".

In termine generico uno schedario, una biblioteca, un vocabolario sono delle memorie.

Nel campo nostro specifico i nastri magnetici, le bande e le schede perforate costituiscono altrettante memorie che, in ossequio ai supporti che contengono i dati, sono dette memorie di input e di output. Esse sono memorie d'archivio in quanto conservano le informazioni per un numero di tempo considerevole, o comunque assai maggiore delle memorie di lavoro o memorie principali.

Le memorie principali rappresentano il punto di smistamento, il passaggio obbligatorio di tutti i dati destinati all'elaborazione.

Inoltre esse contengono registrato il programma, cioè l'insieme di istruzioni scritte in grado di comandare l'esecuzione di un'elaborazione.

Se ci si consente il traslato, possiamo paragonare i due tipi di memorie viste rispettivamente alle corsie e alla sala operatoria di una divisione chirurgica. Le corsie stanno alle memorie di imput e output, la sala operatoria alla memoria di lavoro. E' evidente che, come tutti i ricoverati debbono passare per la sala chirurgica se vogliono essere operati, così tutti i dati di imput devono attraversare la memoria principale, dalla quale usciranno come risultati (output).

3.2 Memorie ad accesso sequenziale e ad accesso casuale

Accedere ad una memoria significa portarsi in una delle sue posizioni per esaminarne o modificarne il contenuto.

I modi, con cui è possibile accedere ad una memoria, sono sostanzialmente due e prendono il nome di accesso sequenziale e accesso casuale.

Si supponga una portineria d'albergo, dove le chiavi delle stanze sono sospese lungo un filo che le attraversa per un foro.

Evidentemente, se il portiere deve prendere una chiave (per esempio la n° 7) per consegnarla ad un cliente, dovrà prima sfilare ad una ad una le chiavi che stanno fra un capo del filo e la n° 7. Mentre, se le singole chiavi fossero appese al filo con un uncino oppure a tanti chiodi, il portiere (sia pure dopo un certo tempo di ricerca), consegnerà la chiave al cliente, sfilando quella soltanto.

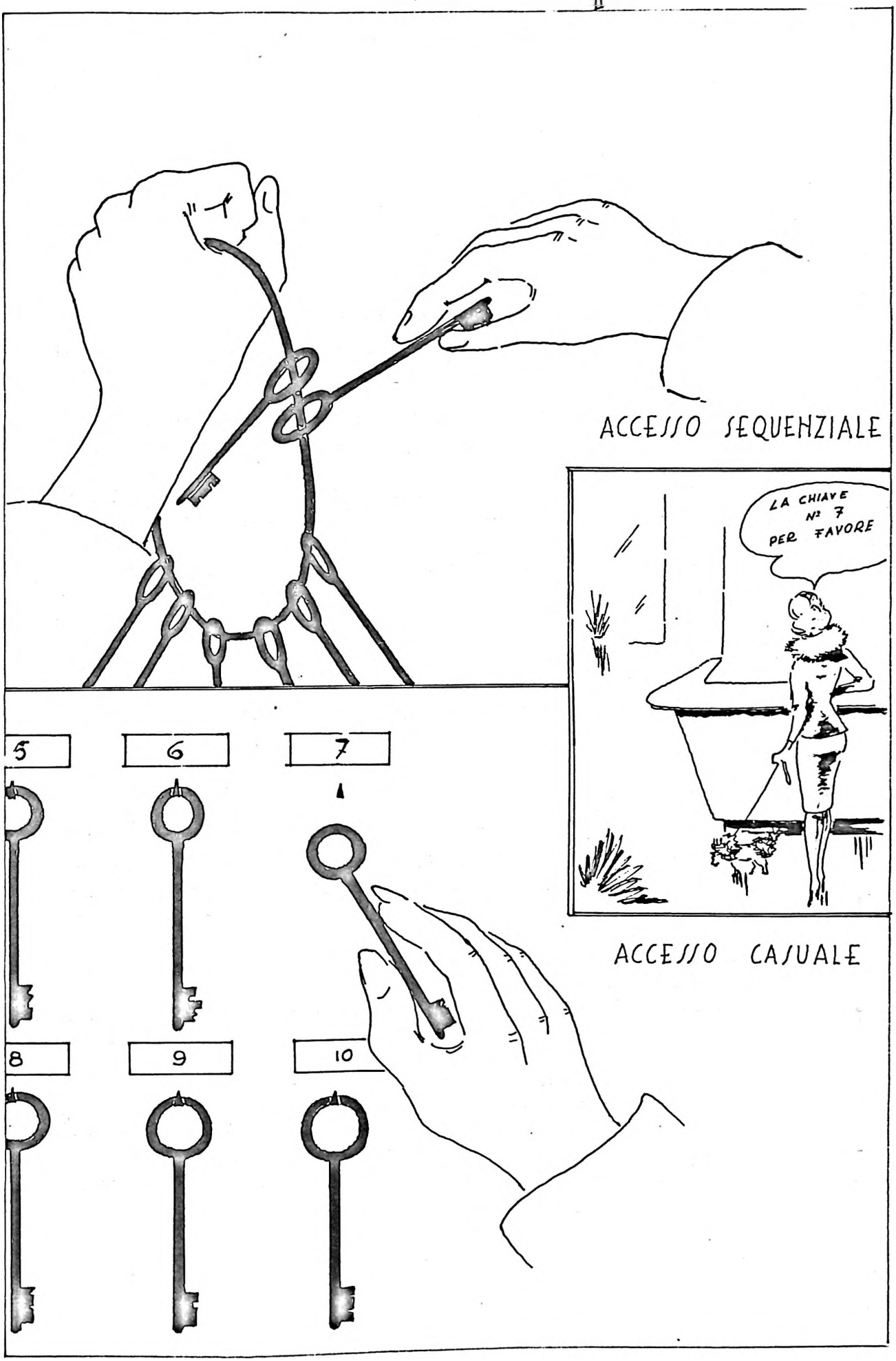
Nel primo caso abbiamo un esempio di accesso sequenziale, nel secondo abbiamo un esempio di accesso casuale.

Analogamente succede per le memorie. Le memorie a nastro sono memorie ad accesso sequenziale; infatti per trovare un dato occorre effettivamente esaminare tutti quelli che lo precedono.

Le memorie a tamburo, le memorie a disco, le memorie a nuclei sono ad accesso casuale, perchè non hanno bisogno di esaminare nessun'altra posizione per ricercare quella desiderata.

Le memorie ad accesso sequenziale per una loro conveniente utilizzazione necessitano di un ordinamento dei dati e delle informazioni in base ad un determinato codice (detto chiave ordinamento) che consente un esame di tutte le posizioni in un unico senso.

FIGURA 3-1


ACCESSO SEQUENZIALE
LA CHIAVE Nº 7 PER FAVORE
ACCESSO CASUALE
5
6
7
8
9
10

3.3 L'indirizzo

Per accedere ad una determinata posizione in modo casuale bisogna essere in grado di identificarla inequivocabilmente.

A questo proposito ad ogni posizione si dà un numero o un nome, in base al quale la posizione resta individuata automaticamente. Tale designazione si chiama indirizzo.

L'indirizzo è dunque il simbolo, attraverso il quale una posizione, e una sola, resta identificata (figura 3.1).

Vi sono memorie che hanno un indirizzo per ogni carattere, in tal caso si diranno indirizzabili "al carattere", oppure "a parola variabile".

Vi sono invece memorie che hanno un indirizzo ad ogni gruppo di caratteri, di lunghezza fissa e determinata, la loro indirizzabilità è pertanto "alla parola" e le memorie così costituite si chiamano "a parola fissa".

3.4 Individuazione di una parola in memoria

Nelle memorie "a parola fissa" il problema dell'individuazione di una parola è molto semplice, basta infatti indicare il semplice indirizzo.

Per le memorie indirizzabili al carattere invece la procedura è diversa in rapporto alle loro caratteristiche costitutive, che vedremo qui di esaminare. Occorre prima di tutto stabilire il senso di lettura o di registrazione.

Vi sono alcune memorie che vengono lette da destra a sinistra, cioè per moduli di indirizzo decrescenti; altre che procedono da sinistra a destra, cioè per moduli di indirizzo crescenti; infine vi sono memorie che procedono in un senso o nell'altro, secondo il tipo di operazioni da eseguire. Osserviamo la figura 3.2a.

per leggere (registrare) la parola di memoria 16007582 , si darà l'indirizzo di partenza nell'esempio 1800, se il senso è decrescente; 1793, se il senso è crescente.

Individuato il carattere di partenza, occorre far proseguire la lettura (registrazione) per tanti caratteri, fino a completare la parola voluta.

I metodi fondamentali sono quattro e ogni particolare memoria ne sceglie uno o alcuni, secondo le sue caratteristiche.

1) Indicare assieme all'indirizzo di partenza quello d'arrivo.
   Es. 1800, 1793 per lettura (registrazione) a senso decrescente;

   1793, 1800 per lettura (registrazione) a senso crescente.

   Questo caso non è molto usato.

2) Indicare assieme all'indirizzo di partenza il numero di caratteri che compongono la parola (lunghezza).
   Es. 1800, 8; oppure 1793, 8.

3) Farla terminare a sinistra (se il senso è decrescente),a destra (se il senso è crescente) con un carattere particolare, detto carattere chiave o sentinella, nell'esempio ! (punto esclamativo). In tal caso, posto l'indirizzo di partenza, si procederà alla lettura (registrazione) fino all'incontro di questo carattere.

4) Accendere in corrispondenza dell'ultimo carattere della parola un bit supplementare, detto bit di fermo, che rimane spento per tutti gli altri caratteri. In tal modo la memoria dovrà avere un bit in più per ogni carattere.
   Una volta posto l'indirizzo di partenza la lettura avverrà fino al carattere dotato di bit di fermo e così pure la registrazione.

FIGURA 3-2 II

a)

Posizione	!	1	6	0	0	7	5	8	2	!
Indirizzo	1792	1793	1794	1795	1796	1797	1798	1799	1780	1781

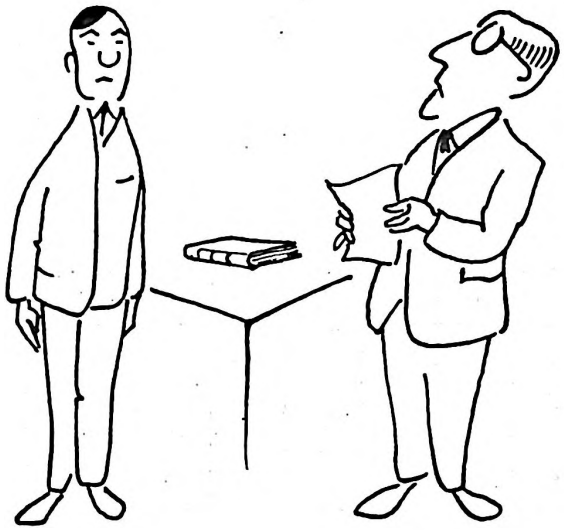
b)

BIT DI FERMO	1	0	0	0	0	0	0	0
BIT DI CONTROLLO	0	1	1	1	0	0	1	1
BIT A	1	1	0	0	1	0	1	1
BIT B	0	0	0	0	1	0	1	1
BIT C	0	0	0	0	0	0	1	0
BIT D	0	1	0	0	1	1	1	0
BIT E	0	0	0	0	0	0	0	0
BIT F	0	0	0	0	0	0	0	0

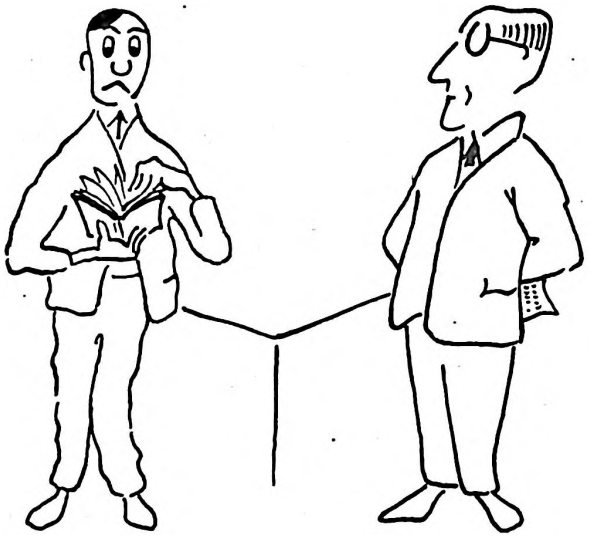
1	6	0	0	7	5	8	2
1793	1794	1795	1796	1797	1798	1799	1780

FIGURA 3-3 II

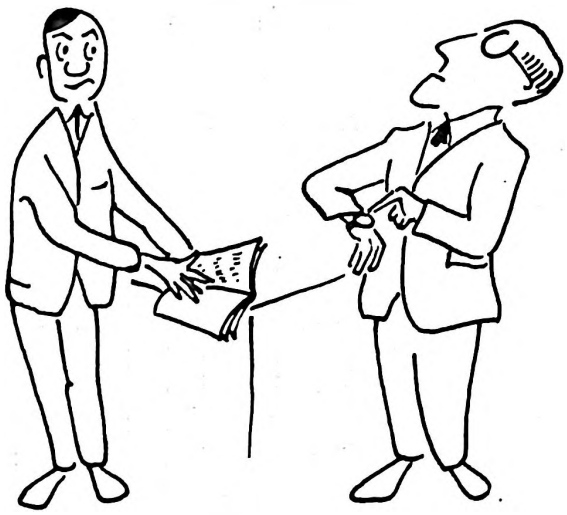
TEMPO D'ACCESSO



1 - Leggere a pagina 170 capoverso 3



2 - Ricerca pagina e capoverso



3 - Lettura

Nell'esempio riportato (figura 3.2-b), partendo dalla posizione 1800 ci si arresterà alla posizione 1793, in corrispondenza della quale il carattere ha bit di fermo con valore 1.

3.5 Il tempo di accesso

Il tempo d'accesso è l'intervallo che intercorre fra l' istante in cui si comanda la richiesta di un dato (per la lettura o la registrazione in memoria) e il tempo in cui il dato viene letto o registrato.

Il tempo d'accesso è scomponibile in due intervalli consecutivi, detti tempo d'attesa e tempo di lettura (o registrazione).

$$T_{acc} = T_{att} + T_{lett}.$$

Infatti, una volta comandata la lettura di una determinata posizione, trascorrerà un intervallo prima di raggiungerla e un intervallo per procedere successivamente alla lettura della posizione stessa.

Si supponga che l'insegnante comandi la lettura del paragrafo di un testo. All'allievo occorrerà un certo periodo di tempo per trovare quel paragrafo e un cert'altro per leggerlo (fig. 3.3).

Il tempo d'accesso costituisce la misura della velocità delle memorie, non solo, ma nel caso delle memorie principali di lavoro il tempo d'accesso dimensiona la velocità di tutto l'elaboratore. Se si pensa infatti alla funzione della memoria di lavoro, che è quella di servire da centro di elaborazione di tutti i dati, risulta evidente come la sua velocità sia determinante per tutto il complesso.

3.6 La circolarità delle memorie

E' circolare tutto ciò che idealmente può essere ricondotto allo schema di una circonferenza.

A questo proposito anche le memorie possono essere circolari. -

Risultano circolari quelle memorie che hanno tutte le posizioni contigue fra di loro in modo, che anche la prima è contigua dell'ultima. Il che equivale a dire che se ci si sposta di un indirizzo dopo l'ultima posizione si è sulla prima e viceversa.

Le memorie circolari non presentano soluzione di continuità negli indirizzi. Non sono cioè una via diritta, ma una piazza a circolo, dove, se ci si sposta in avanti, dopo l'ultima l'ultima casa si incontra la prima.

E' quindi possibile in una memoria, per esempio di 40.000 posizioni, registrare una informazione di 1000 caratteri a partire per esempio dalla posizione 39.700 in avanti. I primi 300 caratteri arriveranno fino a 39.999, gli altri 700, per la continuità delle posizioni, da 0 a 700.

3.7 Modularità delle memorie

Un organo si dice modulare, quando è possibile aumentare o diminuire la sua capacità semplicemente aggiungendo o togliendo un numero di elementi di base (detti moduli), di cui esso risulta composto.

E' questo un requisito importante delle memorie perchè il mutare di determinate condizioni che suggeriscono di variare la loro ampiezza, non implica necessariamente la completa sostituzione dell'organo con uno di dimensioni differenti da parte dell'utente .

3.8 Classificazione dei vari tipi di memoria

Diamo la seguente classificazione delle memorie :

Memorie esterne o memorie lente
- nastro magnetico
- banda perforata
- scheda perforata

Memorie intermedie ausiliarie
- tamburo magnetico
- disco magnetico

Memorie principali di lavoro
- a nuclei magnetici
- a tamburo magnetico

Delle memorie esterne abbiamo ampiamente parlato nel capitolo precedente. Mentre delle memorie ausiliarie diciamo semplicemente che contengono dei dati di consultazione assai più frequente rispetto a quelli registrati nelle memorie di input, per cui la memoria principale può servirsene con maggiore comodità e rapidità.

Oltre a quelle citate vi sono altre memorie, dette buffer, che hanno funzione di smistamento delle informazioni fra unità lente e unità più veloci.

Una ulteriore classificazione può essere fatta, tenendo presente alcuni requisiti particolari, fra memorie labili e memorie persistenti, ricordando che le prime conservano i dati per un periodo di tempo relativamente breve rispetto alle altre.

Vi è poi la distinzione fra memorie dinamiche e memorie statiche, a seconda che il mezzo in cui sono registrati i dati sia un movimento oppure no.

3.9 Memorie a tamburo magnetico

Una memoria a tamburo magnetico è costituita da un cilindro metallico, rivestito da una pellicola di materiale ferromagnetico, che ruota a velocità elevata intorno al suo asse.

Disponendo vicino a questa memoria una testina di registrazione e di lettura, del tutto analoghe a quelle studiate per i nastri è possibile ispezionare la porzione di superficie laterale del cilindro che viene a trovarsi in corrispondenza delle testine durante il tragitto di rotazione. Tale porzione di superficie (che corrisponde alle piste in cui si era suddiviso idealmente il nastro magnetico) è detta pista o banda del tamburo e può presentare le seguenti caratteristiche:

1) registrazione (lettura) del carattere in serie cioè un bit dopo l'altro (figura 3.4-b);

2) registrazione (lettura) del carattere in parallelo e della parola in serie (figura 3.4-c);

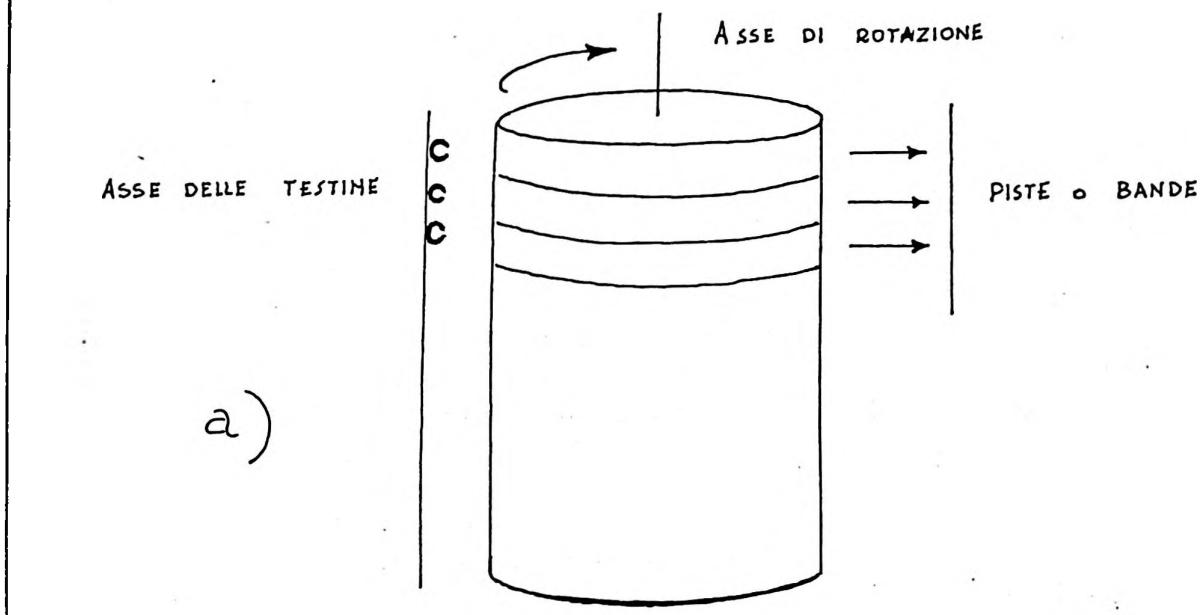
3) registrazione (lettura) del carattere in parallelo e della parola in serie ma per numero di caratteri obbligatoriamente fisso (figura 3.4-d).

Nel primo caso la testina di registrazione (lettura) è evidentemente una sola per pista, mentre nel caso 2 e 3 se ne hanno per pista un intero gruppo pari al numero di bit per carattere.

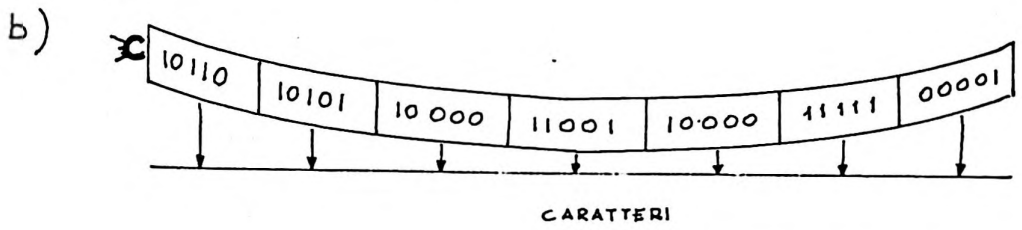
Il tamburo risulta quindi diviso in tante bande parallele ed è una memoria del tipo dinamico, in quanto la registrazione e la lettura avviene mediante rotazione del cilindro sotto le testine. -

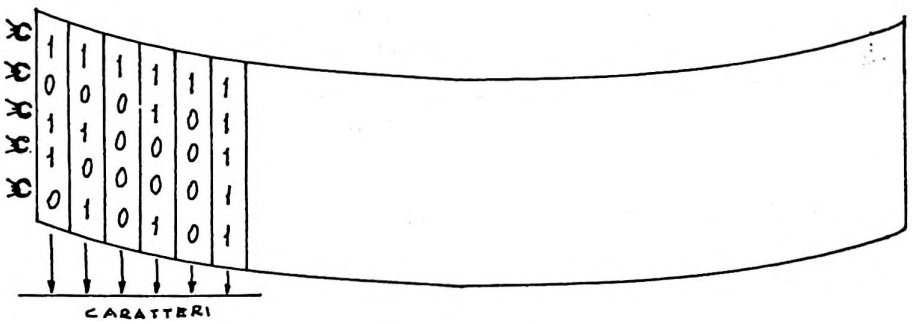
L'indirizzamento in queste memorie è ottenuto mediante indicazione della pista e della posizione relativa all'interno della stessa. Poichè esiste un gruppo di testine ogni pista, l'accesso al-

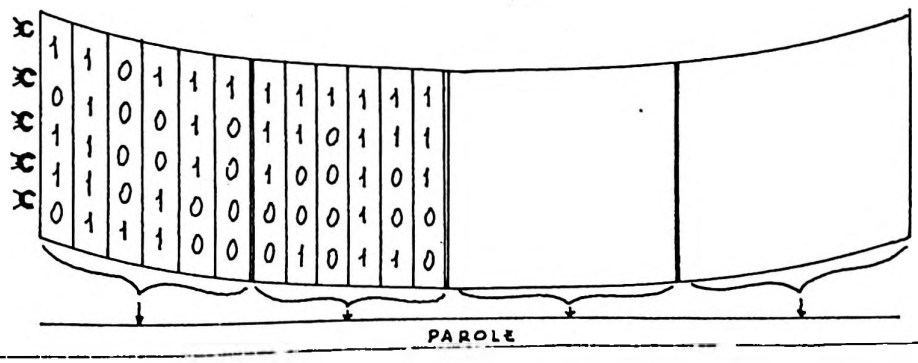
FIGURA 3-4 II



MEMORIA A TAMBURO







la pista è immediato, ciò che non succede per l'esplorazione della sua posizione interna.

Nelle memorie a tamburo si osserva infatti che il tempo di lettura e di registrazione è abbastanza piccolo, fino a raggiungere dimensioni dell'ordine dei microsecondi; mentre il tempo di accesso risulta sempre elevato per la presenza di un sensibile tempo di attesa.

Poichè il tempo d'attesa può variare secondo che la posizione da ricercare sia vicina o lontana dalle testine della pista, è corretto parlare a priori di tempo d'attesa medio, il quale coinciderà con un mezzo giro di tamburo intorno al proprio asse.

Normalmente il tempo d'accesso delle memorie a tamburo, misurato in tempo d'attesa medio,è dell'ordine di qualche decina di millisecondi. Le memorie a tamburo hanno una capacità che oscilla secondo i tipi da 50.000 a 150.000 posizioni circa.

3.10 Memoria a dischi magnetici

E' costituita da un'insieme di dischi sovrapposti, coassiali, opportunamente distanziati l'uno dall'altro, rivestiti in entrambe le facciate da una pellicola di materiale ferromagnetico e solidali ad un albero di rotazione.

La superficie di ogni disco contiene registrati caratteri per tante piste concentriche. A sua volta la facciata di un disco può essere suddivisa in diversi settori, la cui funzione è quella di localizzare meglio le informazioni.

Le facciate dei dischi vengono esplorate da testine di lettura e di registrazione.

Vi sono memorie a dischi che possiedono un solo braccio di esplorazione, per cui l'individuazione di una informazione è ottenuta attraverso l'esecuzione di tre tipi di movimento con evidente appesantimento del tempo d'attesa (figura 3.5-a).

1) movimento verticale (A) per spostare il braccio sul disco;

2) movimento orizzontale (B) " " " " sulla pista;

3) movimento di rotazione per posizionarsi sul settore.

Il tempo d'accesso, misurato in tempo d'attesa medio e dell'ordine di qualche centinaia di millisecondi.

Recentemente sono state realizzate memorie aventi un braccio per ogni disco, il quale consente di eliminare lo spostamento verticale durante l'indirizzamento, vedi figura 3.5-b; vi sono anche memorie con dischi fra loro intercambiabili, che offrono ai dischi la possibilità di servire, sia pure in forma limitata, da memorie di input o di output.

3.11 Memorie a nuclei magnetici

Si tratta di memorie essenzialmente statiche, costituite da fili conduttori disposti a matrice quadrata, nelle cui intersezioni sono inseriti dei nuclei magnetici di ferrite a forma di anello, ulteriormente attraversati da un conduttore diagonale. -

La forma delle memorie a nuclei è quella di tanti piani a reticolo, paralleli fra di loro, quanti sono i bit che entrano nella configurazione di ciascun carattere (figura 3.6-a).

Intersecando perpendicolarmente un piano in un suo bit, si attraversano simultaneamente tutti gli altri, che nel loro insieme configurano un carattere (figura 3.6-b).

3.12 Registrazione di un carattere nelle memorie a nuclei

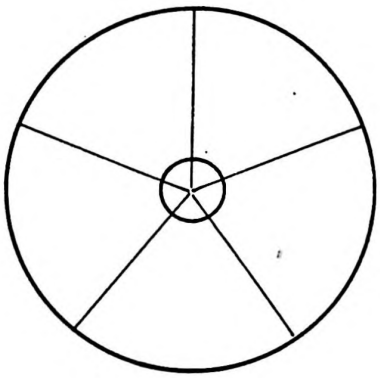
Gli anelli di ferrite possiedono lo stesso ciclo d'isteresi magnetica visto a proposito dei nastri (vedi cap. 2.10).

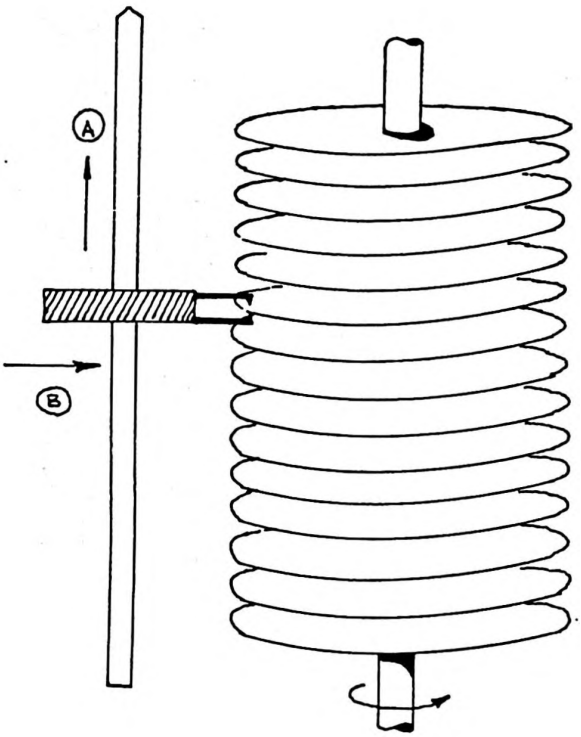
Nel grafico figura 3.6-c si osserva come una corrente $\frac{I}{2}$, sia positiva, sia negativa non modifica per niente il valore del campo ϕ.

FIGURA 3-5

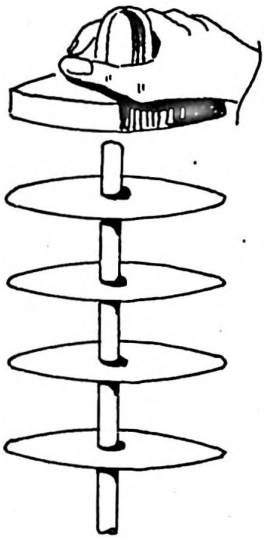
A

FACCIATA DI UN DISCO
A 5 SETTORI





B



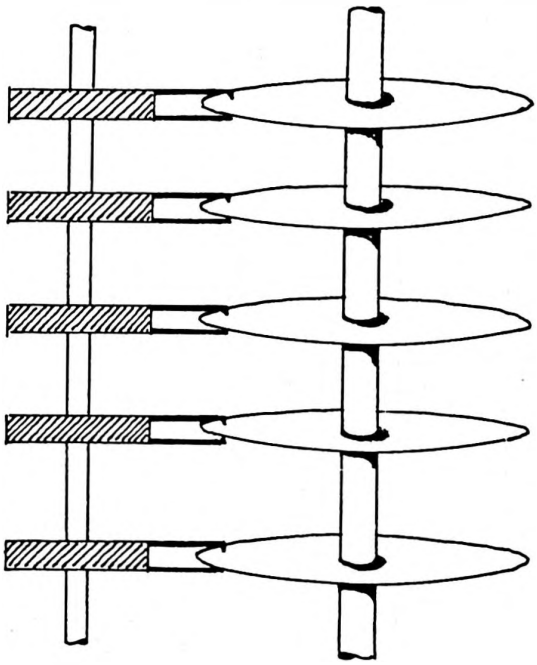
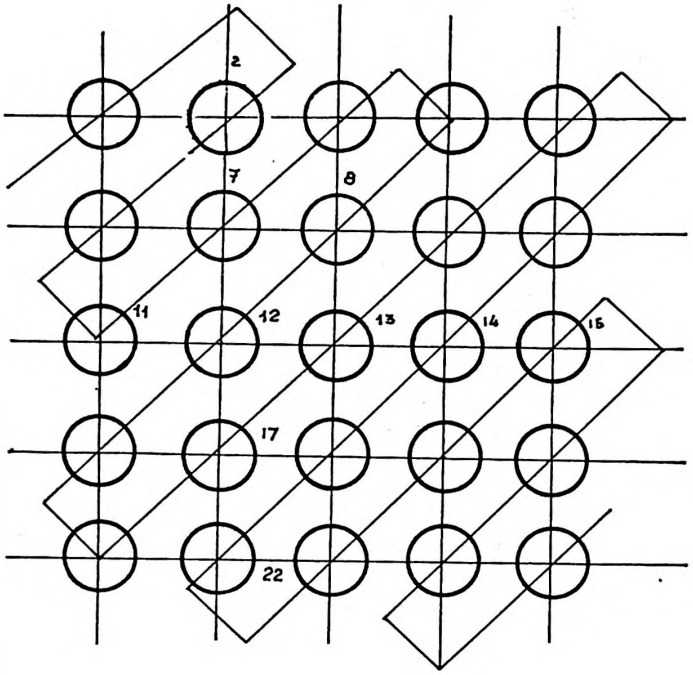
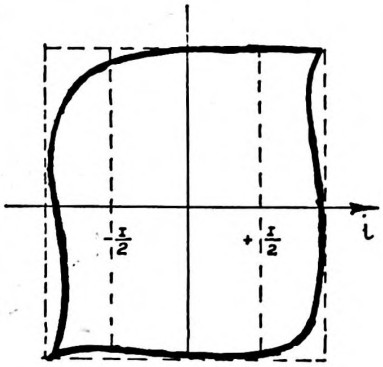
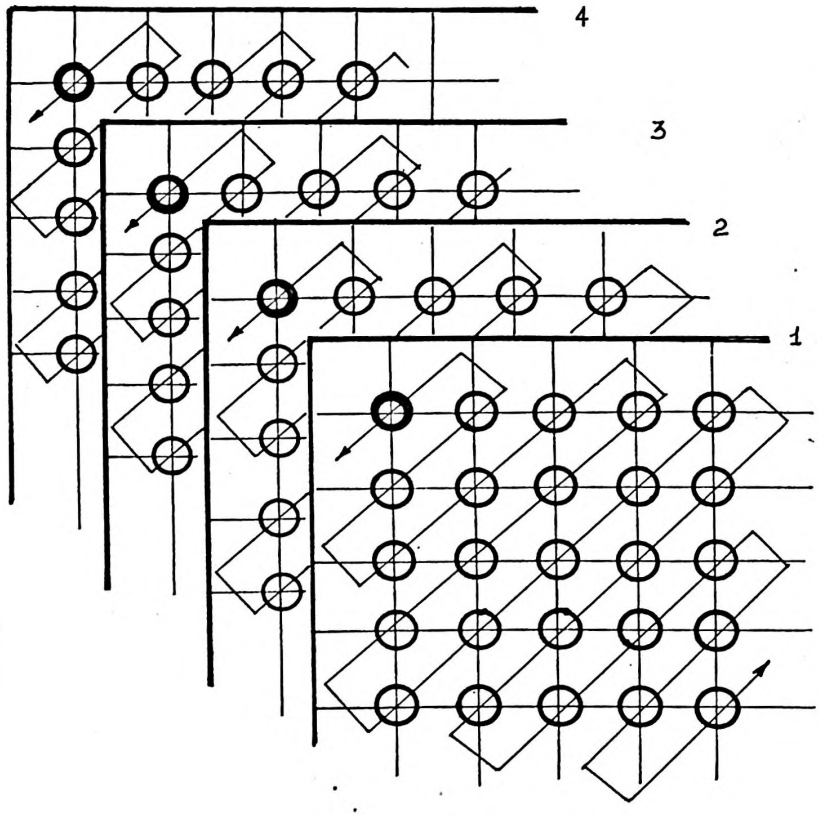
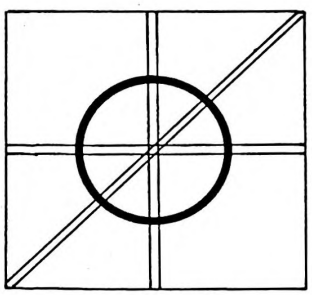


FIGURA 3 - 6









NUCLEO MAGNETICO

Supponiamo di voler magnetizzare positivamente il nucleo posizione 12 (fig. 3. 6 - a) e di far passare una corrente positiva del valore I/2 per il conduttore d'ascissa e I/2 per il conduttore d'ordinata.

E' chiaro che, per il ciclo d'isteresi visto, solo il nucleo 12, intersezione dei due conduttori, rimane influenzato, mentre i nuclei 11, 12, 13, 14, 15 attraversati dalla sola I/2 non mutano il loro stato e non vengono influenzati per niente. Lo stesso di casi per i nuclei 2, 7, 12, 17, 22 attraversati dall'impulso I/2 d'ordinata.

Per questa ragione le memorie a nuclei sono anche chiamate a "coincidenza di correnti".

Volendo magnetizzare in un senso o nell'altro il nucleo di posizione 8 è sufficiente far passare una corrente I/2 (±) sia per il conduttore in ascissa 2 sia per il conduttore in ordinata 3 .

Se invece di un bit si vuole registrare un intero carattere occorrerà far passare simultaneamente sugli stessi conduttori d'ascissa e sugli stessi conduttori d'ordinata, impulsi - I/2 per le configurazioni 0, impulsi +I/2 per le configurazioni 1.

Dopo quanto si è detto è ovvio che vi saranno tanti conduttori di ascissa e d'ordinata quanti sono i piani di bit, cioè quanti sono i bit che concorrono a configurare i vari caratteri più quello di controllo e eventualmente quello di fermo .

3.13 Lettura di un carattere nelle memorie a nuclei

Il nucleo di posizione 12 è magnetizzato in un senso che non conosciamo, dobbiamo stabilire se ha valore 1 oppure 0.

Ebbene, facciamo passare per i conduttori d'ascissa 3 e d'ordinata 2 una corrente -I/2. Se il nucleo 12 era magnetizzato positivamente (valore 1) per il ciclo d'isteresi la corrente del valore -I/2 + (-I/2) = -I lo fa commutare, cioè lo magnetizza in senso opposto (valore 0). La variazione di flusso provocata dalla

commutazione del nucleo, induce attraverso il terzo conduttore un impulso di corrente.

Se invece il nucleo 12 era magnetizzato 0 il valore -I non provoca commutazione e dal conduttore trasversale non passerà nessun impulso di corrente indotta. In questo modo, se passa un impulso attraverso il terzo conduttore possiamo dire che il nucleo era magnetizzato 1, diversamente che era magnetizzato 0.

Se invece di un bit desideriamo leggere un carattere, faremo passare l'impulso -I/2 attraverso tutti i conduttori d'ascissa e di ordinata.

In base a quanto abbiamo detto si rileva che la lettura nelle memorie a nuclei è distruttiva, cioè provoca l'azzeramento di tutti i bit e quindi di tutti i caratteri. Occorrerà pertanto procedere a ripristinare i bit 1 cancellati. Ciò si ottiene facendo ripassare in un tempuscolo successivo una corrente +I/2 per le corrispondenti ascisse e ordinate del carattere.

Vi possono essere alcune memorie a nuclei senza dispositivo di ripristino, questo molto spesso accade per le memorie di transito e di collegamento, dette buffer.

3.14 L'indirizzamento nelle memorie a nuclei

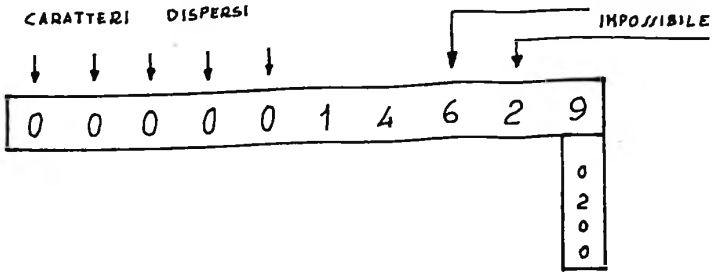
Tecnicamente esso si ottiene decodificando i moduli di indirizzo nei corrispondenti punti di ascissa e di ordinata. Ciò avviene mediante dispositivi che operano a velocità elevatissima dell'ordine di pochi microsecondi, la quale è costante per qualsiasi indirizzo considerato.

Il tempo di ricerca di una qualunque posizione in queste memorie risulta dunque costante e infinitesimo, il che consente di supporre il tempo d'attesa uguale a zero e di identificare il tempo d'accesso col tempo di registrazione - lettura.

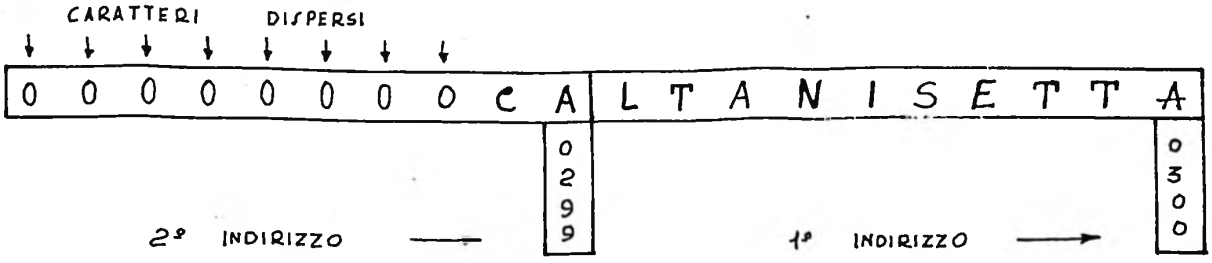
E' questo un requisito importantissimo delle memorie a nuclei , che unitamente alla loro elevatissima velocità di accesso (normalmente dell'ordine di pochi microsecondi) attribuisce loro un

FIGURA 3-7 II

MEMORIA A PAROLA FISSA



MEMORIA A PAROLA FISSA



MEMORIA A PAROLA VARIABILE

NESSUNA DISPERSIONE

1	4	6	2	9
1796	1797	1798	1799	1800

POSSIBILE BASTA INDIRIZZARCI a 1799 PER LUNGHEZZA 2

C	A	L	T	A	N	I	S	E	T	T	A
1889	1890	1891	1892	1893	1894	1895	1896	1897	1898	1899	1900

posto di primo piano negli elaboratori di grande potenza. E' ormai unanime il presupposto che i grandi complessi elettronici debbono avere la memoria di lavoro a nuclei magnetici di ferrite, che per antonomasia sono dette memorie rapide.

Vi sono memorie a nuclei che leggono il carattere in parallelo e la parola in serie con indirizzabilità al carattere; vi sono memorie che leggono un'intera parola fissa in parallelo, in tal caso il loro indirizzo si riferirà ad un gruppo di bit (cella) che individua l'intera parola.

Evidentemente le memorie a parola in parallelo sono più veloci delle memorie in serie, tuttavia mancando dell'indirizzabilità al carattere sono più rigide e dispersive.

Supponiamo infatti una memoria in parallelo a parola fissa di 10 caratteri, si è per forza costretti ad uniformare qualsiasi parola a quella dimensione, anche se la sua lunghezza naturale risulta senz'altro inferiore. Come pure nel caso di una parola più lunga si è costretti ad esprimerla con un numero di caratteri multiplo di 10 e occorre dare più indirizzi. In questo modo si hanno "dispersioni" che in una memoria in serie non avvengono.

Inoltre se si suppone di voler leggere una sola parte della parola, come spesso può accadere, il sistema a parola fissa, data la sua indirizzabilità a gruppi, si presenta assolutamente inadeguato. Vedi figura 3.7.

Queste considerazioni portano a ridimensionare il fattore rapidità nelle memorie a parola fissa. Infatti la maggiore velocità di queste memorie nel confronto con una memoria in serie deve essere calcolata tenendo conto anche del numero di caratteri che mediamente non é utilizzato ogni parola.

E S E R C I Z I

1) Definizione generale di Memoria.

2) Differenza fra memoria d'archivio e memorie di lavoro.

3) Funzione della memoria principale o di lavoro.

4) Differenza fra accesso sequenziale e accesso casuale .

5) Cosa significa dire che una memoria è indirizzabile ?

6) Cos'è una memoria a parola variabile?

7) Come è possibile individuare una parola in una memoria a parola variabile.

8) Cos'è il bit di fermo?

9) Cos'è il tempo d'accesso?

10) Differenza fra tempo di attesa e tempo di lettura/registrazione.

11) Cosa significa dire che una memoria è circolare?

12) Perchè è conveniente una memoria modulare?

13) Quale remora presenta una memoria a dischi e a tamburo, rispetto a una memoria a nuclei?

14) Le memorie a dischi sono ad accesso casuale?

15) Perchè le memorie a nuclei sono dette a coincidenza di correnti?

16) E' più veloce una lettura del carattere in serie o in parallelo?

17) Le memorie a parola fissa quali inconvenienti presentano?

18) La memoria a nuclei è una memoria dinamica?

19) Poichè la memoria a nuclei non ha testine, come avviene l'operazione di lettura?

20) Come si ripristina il carattere cancellato in una memoria a nuclei?

IV GLI ALTRI ORGANI DELL'UNITA' CENTRALE

4.1 Unità aritmetica e logica

E' l'organo tipicamente elaborante dei calcolatori, che consente la trasformazione effettiva dei dati, a seguito di operazioni di tipo aritmetico o di tipo logico (secondo l'algebra di Boole), in risultati d'elaborazione.
Supponiamo di avere una calcolatrice meccanica con i totalizzatori a ruote dentate, questi totalizzatori sono delle specie di memorie di cumulo che si caricano di volta in volta dei risultati delle operazioni.

Le barre dentate sono gli organi di trasmissione che comandano l'esecuzione dell'operazione sulle ruote del totalizzatore. Dette barre e i cinematici ad esse connessi, costituiscono la parte operativa della calcolatrice.

Lo stesso accade per un Elaboratore Elettronico, dove la funzione operativa è assolta dall'Unità aritmetica e logica e quella di totalizzazione dall'Accumulatore.

Non staremo ad esaminare minutamente la struttura tecnica della U.A.L., anche perchè essa varia moltissimo da Elaboratore a Elaboratore e soprattutto perchè sarebbe superflua ai fini del presente manuale. Ci interessa invece vedere le operazioni automatiche che essa è in grado di compiere, le quali si possono riassumere nelle seguenti:

1) operazioni aritmetiche. Si tratta delle operazioni di somma e di sottrazione alle quali si aggiunge la moltiplicazione, mentre invece la divisione non sempre rientra nelle possibilità dell'U.A.L., in quanto si preferisce eseguirla mediante un sottoprogramma (vedi parte 3ª).
Le operazioni possono essere anche algebriche. Se il risultato dell'operazione supera una determinata capacità o lunghezza, può accadere che sia utile segnalare al Governo della macchina tale "straripamento". Ciò avviene mediante una segnalazione binaria, detta appunto overflow.
Come tutte le calcolatrici di qualsiasi tipo e livello, l'UAL

non tiene conto normalmente nei numeri della virgola, intesa come segno di separazione fra cifre intere e decimali.
L'U.A.L. opera come se le cifre componenti il numero fossero tutte intere, spetterà al programmatore sistemare i valori decimali (inserendo eventualmente la virgola), in modo da poterli poi mettere in evidenza negli stampati.

2) Operazioni logiche. Sono le operazioni dell'algebra di Boole viste nella parte 1^ cap. 3°. Esse vengono applicate ai singoli bit che compongono il carattere, e possono essere di tipo diretto o negato. Esempio: somma logica diretta (riunione) fra A e T

A in codice Elea 9003 è 0 1 0 0 0 1
T in codice Elea 9003 è 1 0 0 0 1 1

il risultato è 1 1 0 0 1 1 cioè K.

3) Operazioni di confronto. Mediante queste operazioni è possibile fare la comparazione fra due valori, rispettivamente scelti come primo e secondo termine di paragone.
Il risultato del confronto è una segnalazione binaria (normalmente ottenuta col posizionamento di un flip-flop) del tipo $=, \neq, >, <$ (uguale, diverso, maggiore, minore).
Il risultato è riferito al primo termine del confronto e offre al Governo la possibilità di prendere delle decisioni logiche.

4) Operazioni di trasferimento. Loro funzione è quella di trasferire un dato da un organo ad un altro, con azzeramento e meno delle posizioni di prelievo.

5) Operazioni di ricerca. Consistono nel ricercare in una memoria la posizione di un carattere o di una parola.

Ciascuna U.A.L. effettua poi alcune operazioni particolari ad un determinato tipo di elaboratore, quali ad esempio nell' Elea 9003 e 6001 la modifica automatica delle istruzioni.

4. 2 L'Accumulatore

Non sempre si tratta di un organo distinto della memoria principale, in quanto a volte è una zona della stessa memoria adibita a tale scopo. Quindi più che di un organo tecnologicamente distinto, si tratta di una particolare funzione che consiste nel contenere di volta in volta i risultati delle operazioni aritmetiche e algebriche elaborati dall'U.A.L. La similitudine fatta precedentemente colle rotelle del totalizzatore di una calcolatrice mosse e sollecitate dalle barre dentate, è di per sè sufficiente a chiarire gli aspetti di questa funzione.

4.3 Indirizzatore di programma

Anticipiamo una nozione che sarà ampiamente svolta nella parte 3ª di questo manuale: il programma è una sequenza di comandi impartiti alla macchina, in modo da farle eseguire un certo lavoro, i quali vanno sempre registrati in memoria principale.

La successione di comandi è una "filastrocca" pressapoco riconducibile alla seguente:

1) Vai al tal operando

2) Sommalo col tal'altro.

3) Metti il risultato al posto tale.

4) Confronta detto risultato con l'operando tale.

5) Se è maggiore vai al comando 8 ecc.

La macchina, anzi, il suo governo, procede in questo modo:

- legge il comando 1, lo esegue;
- legge il comando 2, lo esegue;
- legge il comando 3, lo esegue; e così via.

I comandi sono parole strane (chiamate istruzioni), le quali saranno situate in una determinata posizione di memoria, che di vol

ta in volta dovrà essere comunicata alla macchina, allo scopo di farle leggere il comando desiderato.

Tale funzione è assolta dall'indirizzatore di programma, detto anche indirizzatore di istruzioni, che segnala volta per volta al Governo l'istruzione da leggere e da eseguire. Ciò può avvenire in due modi diversi:

1) Ciascuna istruzione contiene indicato l'indirizzo della successiva, che andrà appunto a caricarsi nell'indirizzatore, una volta letta l'istruzione.

2) Le istruzioni sono registrate in memoria una accanto all' altra e vengono eseguite in questo ordine, salvo casi particolari. Ciò permette all'indirizzatore di aggiornarsi automaticamente; questo modo di procedere é detto sequenziale.

4.4 Indirizzatore di operando

L'esecuzione dell'istruzione realizza l'elaborazione di determinati operandi, i quali sono obbligatoriamente situati in memoria principale o in accumulatore.

L'istruzione (secondo le modalità viste al capitolo 3.4) contiene l'indirizzo e le altre indicazioni necessarie a individuare il dato da elaborare, che letti dal Governo Centrale, vengono ricordati da un organo particolare, chiamato indirizzatore di operando. La sua funzione consiste nel segnalare di volta in volta al Governo della macchina la posizione e l'eventuale lunghezza dell'operando su cui elaborare.

Si richiama l'attenzione sui termini di significato ben diverso, onde evitare equivoci dovuti ad una allitterazione casuale.

"Indirizzo dell'istruzione" è la posizione a cui si trova l'istruzione stessa e che andrà nell'indirizzatore di programma.

"Indirizzo contenuto nell'istruzione" è l'indirizzo dell'operando che si intende elaborare coll'istruzione e che interesserà lo indirizzatore di operando.

4.5 Organo di temporizzazione.

E' detto anche "timer" o "clock". E' infatti una specie di orologio, al ritmo del quale sono cadenzate tutte le operazioni che si svolgono nell'Elaboratore.

L'unità di misura di questo organo è il tempo base (detto anche periodo di cifra) e ogni operazione è multipla di questo tempo base.

Vi possono essere più tempi base suggeriti, da particolari circostanze di macchina, in tal caso ogni operazione sarà multipla dei diversi tempi base che intervengono a sincronizzarla.

Il tempo base coincide normalmente coll'intervallo necessario ad operare un bit.

4.6 Unità di Governo

Costituisce ciò che comunemente si chiama il "cervello" dell'Elaboratore.

Come nel corpo umano il "cervello" ha una funzione di coordinamento, di guida e di controllo dei movimenti, così l'Unità di Governo per l'Elaboratore.

I compiti principali del governo nello svolgimento di una operazione sono infatti i seguenti:

- Lettura in memoria delle istruzioni da eseguire.
- Interpretazione del comando d'istruzione.
- Predisposizione e avvio degli organi dell'Elaboratore atti a compiere l'operazione.
- Decisioni di carattere logico a seguito di segnali particolari provenienti dagli organi di elaborazione.
- Controllo di tutta l'esecuzione dell'operazione.

L'importanza dell'Unità di Governo è pertanto fondamentale e si può senz'altro dire che è l'organo responsabile dell'intero andamento dell'elaborazione.

4.7 I canali di collegamento

Il collegamento degli organi dell'Unità Centrale, è ottenuto mediante uno o più canali di smistamento, attraverso i quali durante lo svolgimento dell'elaborazione avviene il passaggio dei caratteri.

Quando i canali sono più d'uno nell'Unità centrale è possibile realizzare un forte risparmio di tempi di esecuzione di uno stesso comando di macchina (istruzione)

Quando invece si vuole ottenere una sovrapposizione di tempi, attraverso lo svolgimento contemporaneo di due o più istruzioni diverse, è necessario inserire degli organi di governo supplementari, che garantiscano lo svolgimento di tali operazioni simultanee.

Per esempio nell'Elea 9003, impiegando per una stessa operazione interna sia il canale interno che quello esterno, si dimezzano i tempi d'esecuzione di alcune operazioni. Sempre in questo Elaboratore, grazie alla presenza del Governo delle Unità a Nastro, è possibile utilizzare simultaneamente: il canale interno, controllato dal Governo Centrale e il canale esterno controllato dal G.U.N. per l'esecuzione di un'operazione interna e di una operazione a nastro!

E S E R C I Z I

1) Quali operazioni normalmente esegue l'U.A.L. ?

2) Quale differenza c'è fra la Memoria e l'U.A.L.?

3) Funzione dell'Accumulatore.

4) Necessità dell'indirizzatore d'istruzioni.

5) Quali modalità di funzionamento può avere l'indirizzatore d'istruzioni?

6) Differenza fra indirizzo dell'istruzione e indirizzo contenute nell'istruzione.

7) Cos'è l'indirizzatore di operando?

8) Cos'è il periodo di cifra?

9) Elencate le funzioni del Governo Centrale.

10) Cos'è la simultaneità operativa?

11) Quale simultaneità realizza la presenza di due canali e un solo Governo?

12) Quale simultaneità operativa consente la presenza di due canali e due Governi?

5 LE ISTRUZIONI

5.1 Definizione

L'istruzione è un comando in base al quale la macchina è in grado di eseguire una operazione di importanza definita.

Si tratterà ovviamente di un comando che già esiste nella logica funzionale dell'Elaboratore, cioè di un comando che esso è in grado di interpretare e quindi di eseguire, che assolutamente non deve essere fuori delle sue possibilità operative. Se la macchina ad esempio manca nell'U.A.L. del dispositivo di risoluzione automatica della divisione, è ovvio che non esisterà un'istruzione specifica per questo tipo di operazione.

Da ciò si deduce che le istruzioni risultano definite e determinate, in rapporto ad un elaboratore particolare, per cui non si può tracciare una linea operativa di macchina senza conoscere le istruzioni che ad essa fanno capo.

Comunque sia tali istruzioni saranno sempre in funzione delle operazioni che l'U.A.L. è in grado di eseguire e delle decisioni logiche che il Governo è in grado di prendere.

5.2 Struttura di una istruzione

Ogni istruzione è strutturata in modo da occupare il minimo numero di caratteri possibile.

Ovviamente se la memoria è a parola fissa l'istruzione verrà dimensionata alla parola.

Mentre per la memoria a parola variabile, vi possono essere Elaboratori con istruzioni a numero fisso di caratteri ed Elaboratori con istruzioni a numero di caratteri variabile.

5.3 Registrazione delle istruzioni

Tutte le istruzioni vengono registrate in memoria principale, dove successivamente verranno lette ed eseguite dalla unità di Governo.

Vi sono Elaboratori, dove la sequenza di istruzioni o programma deve essere registrata in un punto obbligato della memoria, mentre per altri Elaboratori il programma può essere registrato a partire da qualsiasi punto della memoria. E' questo il caso degli Elaboratori Olivetti.

5.4 Componenti dell'Istruzione

Gli elementi di cui si compone un'istruzione si possono suddividere in essenziali e particolari.

I primi sono comuni a tutti gli elaboratori, gli altri sono soltanto peculiari a taluni di essi.

Gli elementi essenziali sono due :

1) Il codice di funzione o codice di operazione. E' il carattere (o i caratteri) in grado di indicare al Governo il tipo di operazione che si desidera eseguire. Per esempio nell'Elaboratore Olivetti Elea 9003 il carattere F significa: trasferisci dalla Memoria in Accumulatore; la stessa operazione nell'Elea 6001 è espressa coi caratteri: 10;

2) Elementi che valgono ulteriormente a precisare il comando espresso nel codice di funzione.

Supponiamo il comando di trasferimento dalla Memoria in Accumulatore.

E' chiaro che occorrerà specificare sia l'operando di memoria che di accumulatore. Anche se l'accumulatore è di una sola parola, occorrerà sempre specificare l'operando di memoria nei modi visti al capitolo 3.4, secondo che la memoria è a parola fissa o a parola variabile.

Supponiamo un comando che ricerca in memoria la posizione di un carattere.

Quale carattere? Occorrerà precisarlo nell'istruzione. A partire da quale punto della memoria? Ci vuole dunque un indirizzo di partenza da cui iniziare la ricerca.

Gli elementi particolari possono invece essere diversi e ovviamente riguardano il tipo specifico di Elaboratore a cui si fa riferimento.

Tali sono ad esempio l'indicazione del registro di modifica per le macchine che hanno la modifica automatica delle istruzioni e l'indicazione dell'indirizzo dell'istruzione successiva per le macchine che leggono le istruzioni non nello stesso ordine in cui sono registrate in memoria.

5.5 Indirizzo dell'istruzione

Già se ne è parlato a proposito dell'indirizzatore programmi. -

Vogliamo qui soltanto rilevare che nel caso di Memoria a parola variabile a indicare l'indirizzo dell'istruzione è sufficiente lo indirizzo del primo carattere del codice di funzione.

5.6 Fase preparatoria e fase esecutiva

Evidentemente, prima di eseguire un comando occorre interpretarlo, e, giacchè per un'istruzione si tratta di un comando registrato in memoria, occorre anche leggerlo.

La fase di lettura e interpretazione di un'istruzione eseguita dal governo è detta fase preparatoria.

Negli Elaboratori Olivetti è detta pure fase α .

Qualunque sia il tipo di Elaboratore interessato, durante la fase preparatoria si svolgono le operazioni seguenti:

1) lettura del codice di funzione e relativa percezione del tipe di comando da eseguire. Tutto ciò avviene mediante opportuni decodificatori di funzione del governo, che simultaneamente provvedono ad abilitare gli organi interessati all'esecuzione successiva, es. Memoria, Accumulatore, eco.

2) operazioni che consentono al governo di individuare l'operando (caricamento dell'omonimo indirizzatore) e di arrestare l'operazione ad esecuzione avvenuta, passando all'istruzione successiva.

Durante la fase esecutiva, che negli Elaboratori Olivetti è detta pure fase β, avviene l'effettivo svolgimento dell'operazione sotto il controllo del Governo.

E S E R C I Z I

1) Cos'è un'istruzione?

2) Quale relazione deve esistere fra le istruzioni e la logica funzionale di macchina?

3) Cos'è il cidice di funzione?

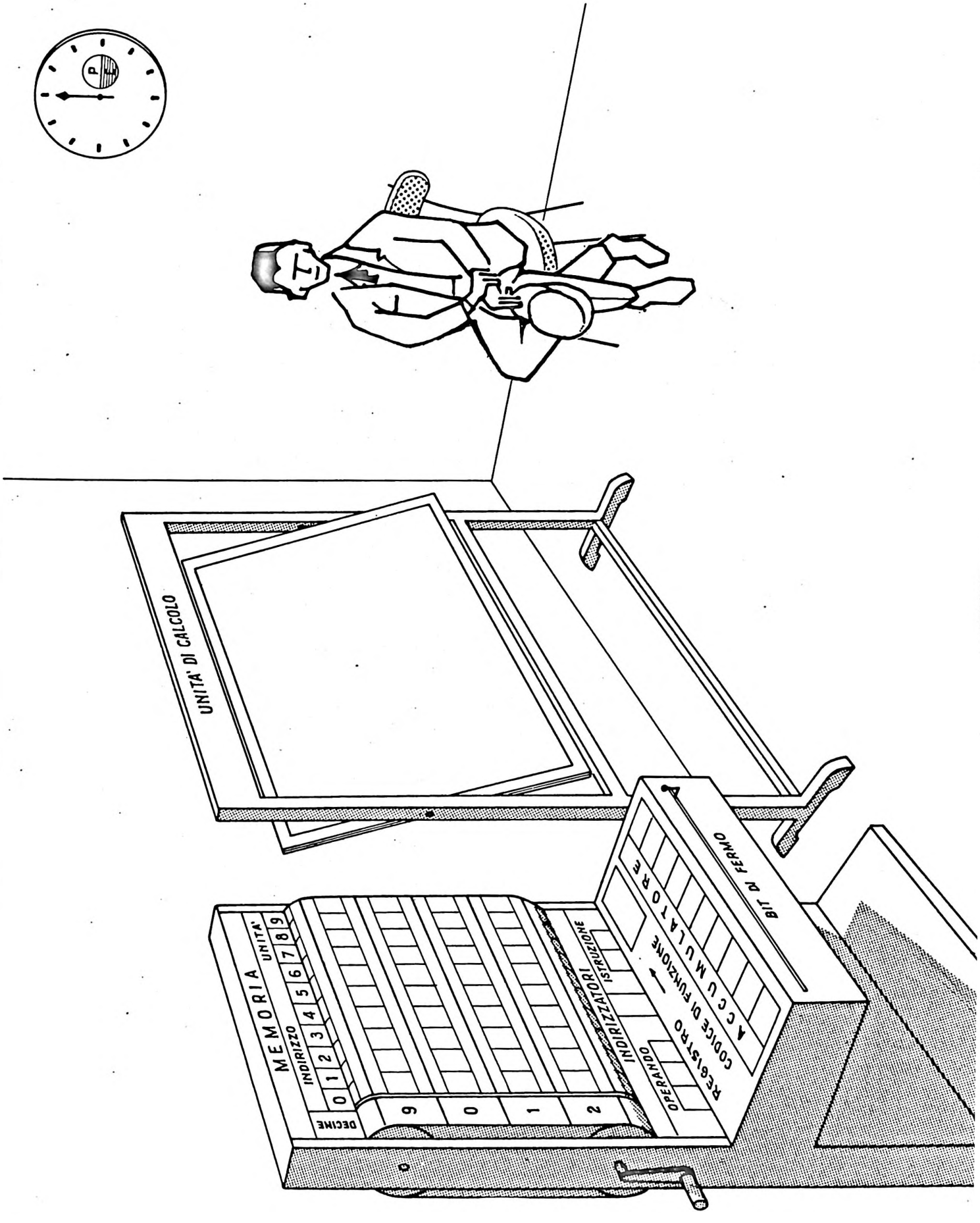
4) Significato di fase preparatoria e fase esecutiva.

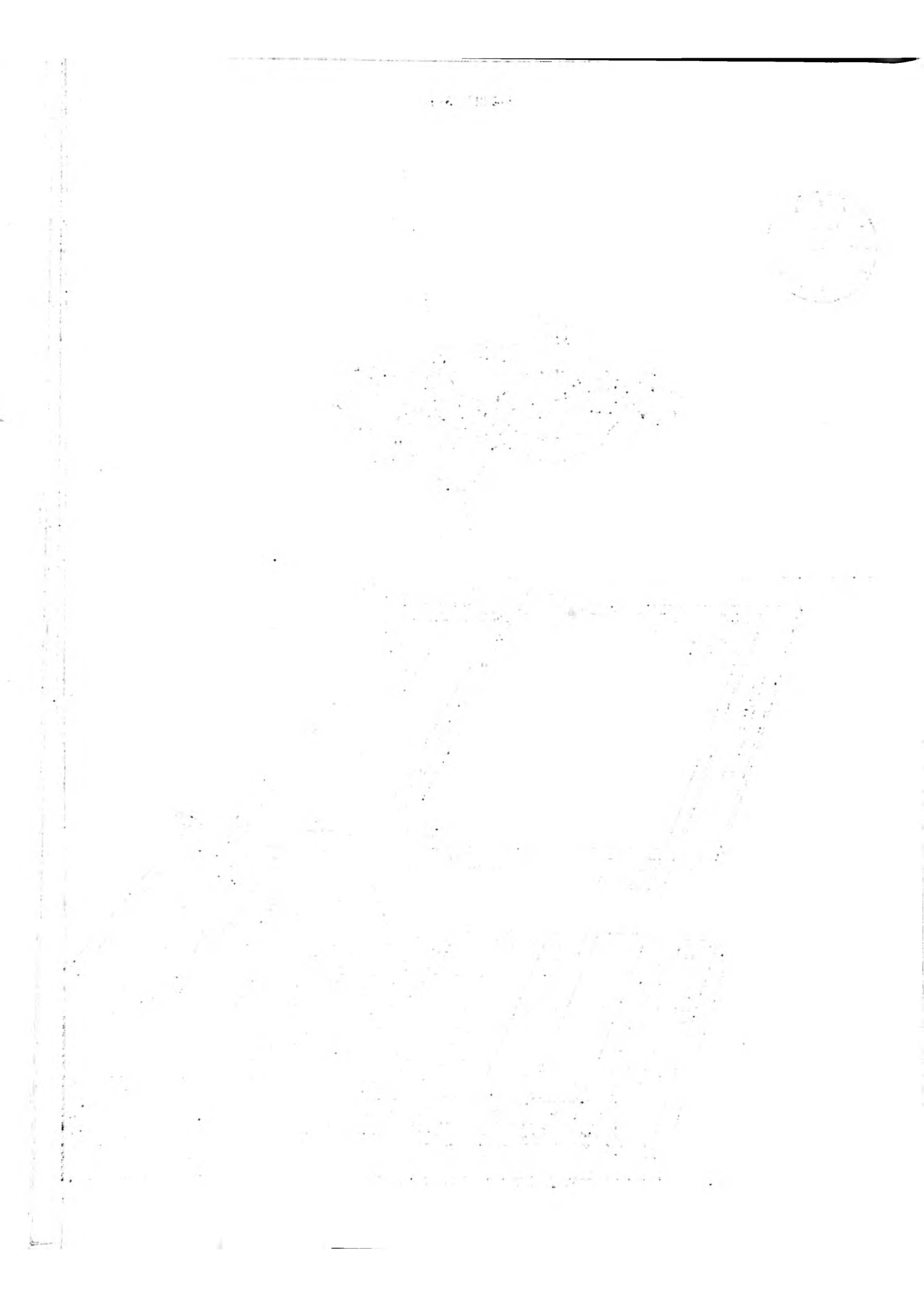
5) Quale organo legge le istruzioni?

6) Deve sono registrate le istruzioni di programma?

7) Quali operazioni particolari avvengono durante la fase preparatoria?

FIGURA 6-1


UNITA' DI CALCOLO
MEMORIA
INDIRIZZO
0 1 2 3 4 5 6 7 8 9
UNITA'
DECINE
INDIRIZZATORI
ISTRUZIONE
REGISTRO
CODICE DI FUNZIONE
ACCUMULATORE
OPERANDO
BIT DI FERMO
9 0 1 2



VI SVOLGIMENTO DI ISTRUZIONI IN UN ELABORATORE IDEALE

Nel corso di questo capitolo illustreremo le modalità con cui un Elaboratore esegue una sequenza di istruzioni.

A tale scopo si è ipotizzato l'Elaboratore illustrato in fig. 1 e dotato delle caratteristiche generali seguenti:

1) Memoria principale a nuclei magnetici di 100 posizioni (numerate da 01 a 00), circolare e indirizzabile alla singola posizione. Le parole sono individuate, a partire dalla posizione del primo carattere di destra, dalla lunghezza espressa numericamente.

2) Accumulatore, distinto dalla memoria principale, a indirizzo unico, della capacità massima di 10 caratteri, dove la presenza del bit di fermo contrassegna la fine della parola. Cioè le parole in accumulatore si estendono, dalla prima posizione di destra fino al bit di fermo.

3) Istruzioni sempre di 4 caratteri, ciascuno dei quali ha questo significato:

   il 1° carattere di sinistra è il codice di funzione;

   il 2° e il 3° carattere indicano l'indirizzo di destra, dell'operando di memoria;

   il 4° carattere definisce il numero di caratteri, (lunghezza) di cui si compone l'operando stesso.

4) Indirizzatore di istruzioni.

5) Indirizzatore di operando: 2 caratteri per l'indirizzo, 1 carattere per la lunghezza.

6) Unità di calcolo, mediante la quale si effettuano tutte le operazioni.

7) Orologio numerato a periodi di cifra. Il quadrante piccolo indica nella parte superiore bianca la fase preparatoria dell'istruzione, nella parte inferiore ombreggiata la fase esecutiva.

Il tempo di preparazione è di 4 periodi di cifra, quello di esecuzione è pari a tanti periodi di cifra quanti sono i caratteri del risultato dell'elaborazione.

Supponiamo di voler eseguire un breve programma, composto dalle operazioni seguenti :

MA = Porta A in accumulatore.

+MA = Somma Acc. con B, risultato D in accumulatore.

-MA = Sottrai C da D, risultato D in accumulatore.

AM = Porta D in zona E di memoria.

STOP = Stop

A è in memoria a partire dall'indirizzo 12 per lunghezza di 3 caratteri (cioè occupa le posizioni 10, 11, 12);

B è in memoria dall'indirizzo 18 per la lunghezza 6 caratteri;

C è anch'esso in memoria dall'indirizzo 22 per lunghezza 4;

E andrà in memoria dall'indirizzo 28 per lunghezza 6.

Le istruzioni che consentono di svolgere tale programma, risultano codificate in questo modo :

Indirizzo istruzione	codice di funzione	indirizzo operando	lungh.	cod. mnem.	DESCRIZIONE
90	F	12	3	MA	porta A Acc
94	C	18	6	+MA	Acc + B Acc
98	H	22	4	-MA	Acc - C Acc
02	G	28	6	AM	Acc E
06	0	00	0	STOP	Stop

Questo programma viene introdotto in memoria, a cominciare dall'indirizzo 90, insieme con i dati da elaborare come rappresenta la figura 6.2.

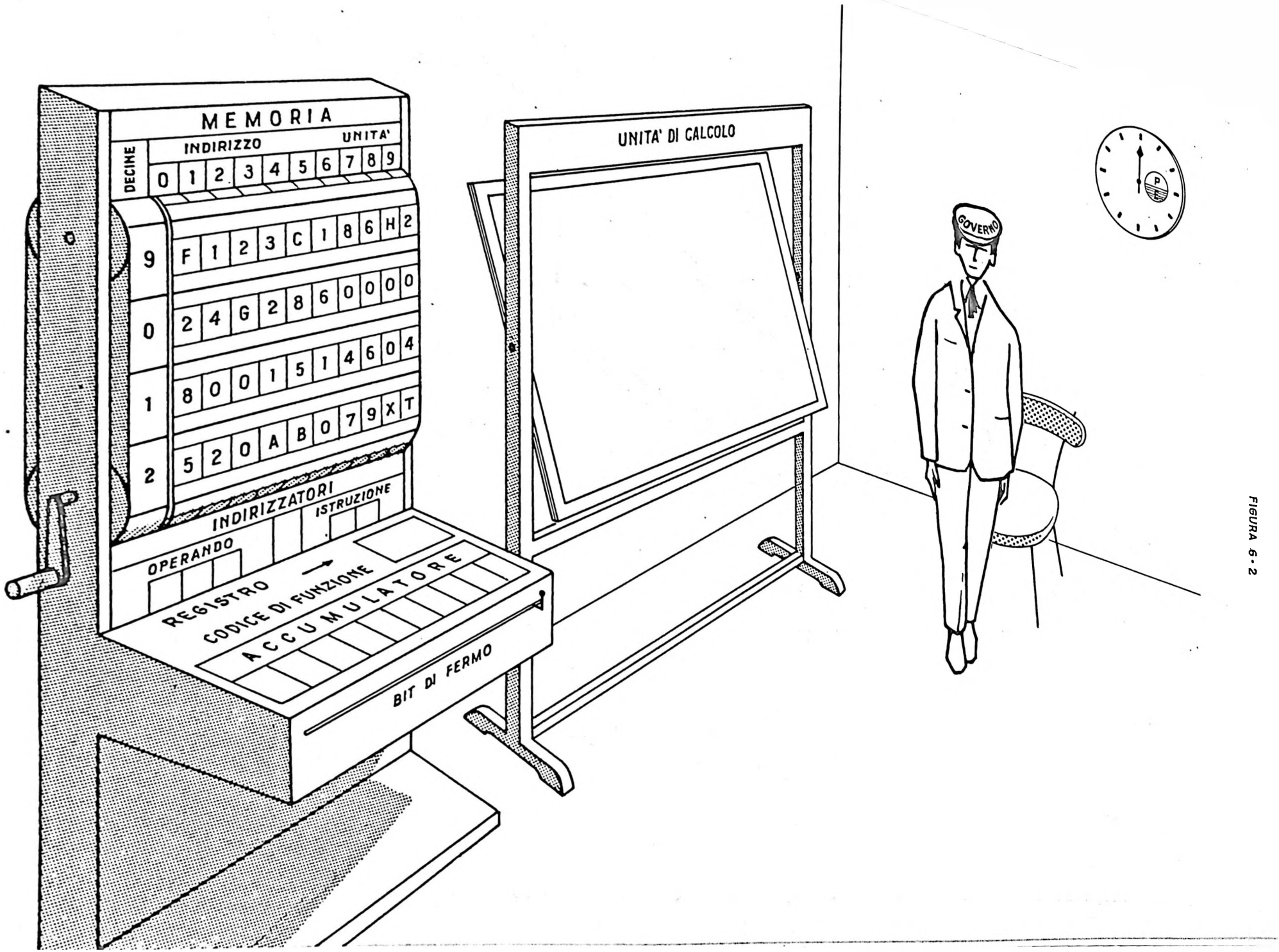

MEMORIA
INDIRIZZO
UNITA'
DECINE
0 1 2 3 4 5 6 7 8 9
9 F 1 2 3 C 1 8 6 H 2
0 2 4 G 2 8 6 0 0 0 0
1 8 0 0 1 5 1 4 6 0 4
2 5 2 0 A B 0 7 9 X T
INDIRIZZATORI
ISTRUZIONE
OPERANDO
REGISTRO
CODICE DI FUNZIONE
ACCUMULATORE
BIT DI FERMO
UNITA' DI CALCOLO
GOVERNO

FIGURA 6 - 2

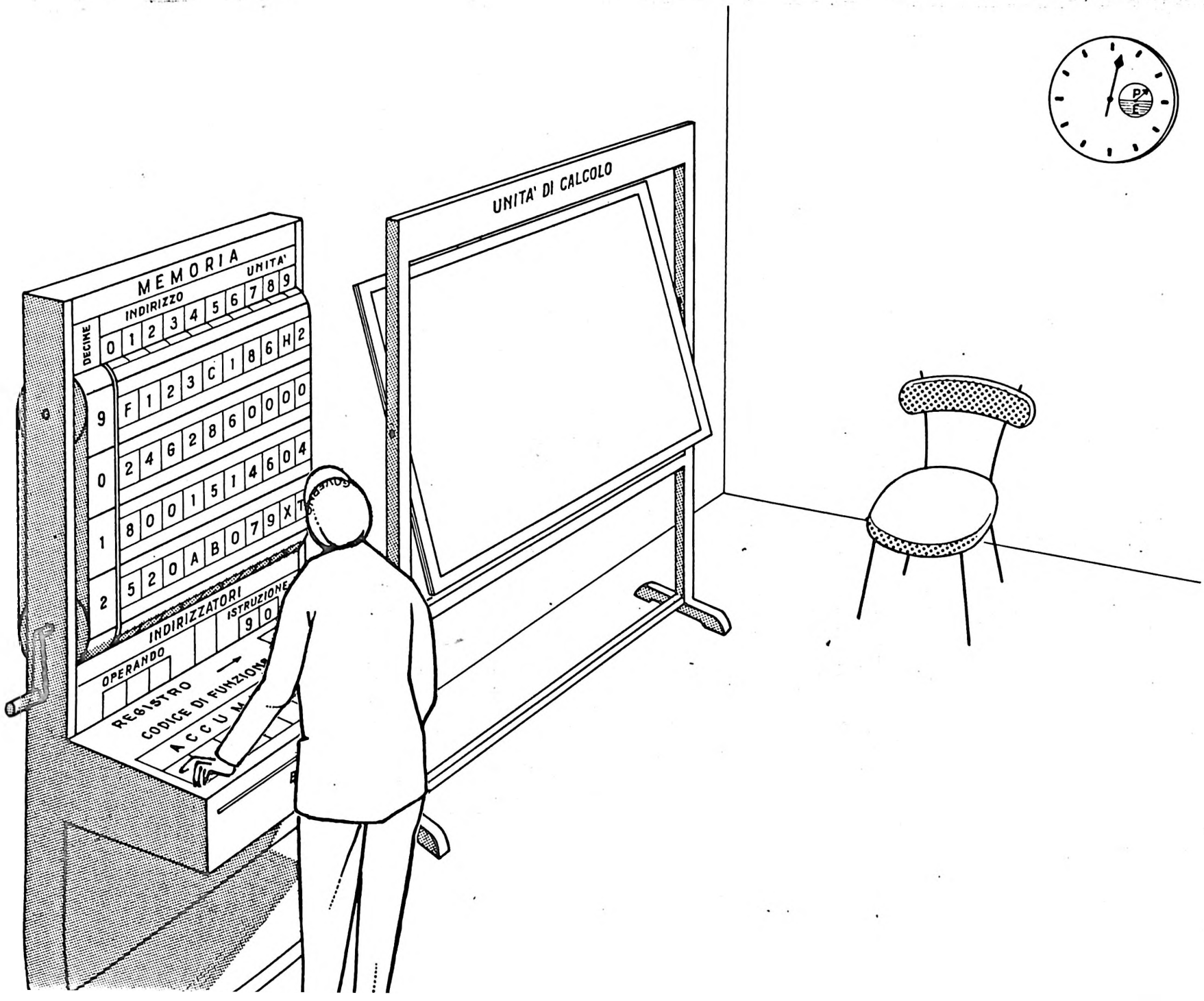

MEMORIA
INDIRIZZO
UNITA'
DECINE
0 1 2 3 4 5 6 7 8 9
9 F 1 2 3 C 1 8 6 H 2
0 2 4 6 2 8 6 0 0 0 0
1 8 0 0 1 5 1 4 6 0 4
2 5 2 0 A B 0 7 9 X
INDIRIZZATORI
ISTRUZIONE
9 0
OPERANDO
REGISTRO
CODICE DI FUNZIONE
A C C U M
UNITA' DI CALCOLO

FIGURA 6·3

La fig. 6,3 mostra l'inizio della fase preparatoria :

Nell'indirizzatore istruzioni è registrato l'indirizzo iniziale della 1ª istruzione.

Fig. 6.4 : conosciuto l'indirizzo iniziale di programma, il governo trascrive: il codice di funzione nell'apposito registro i caratteri indicanti l'indirizzo e la lunghezza nell'indirizzatore di operando. Via via che i caratteri vengono prelevati l'indirizzatore di memoria progredisce in modo che alla fine, esso contiene l'indirizzo iniziale dell'istruzione successiva, cioé 94.

FIGURA 6·4

UNITA' DI CALCOLO
MEMORIA
INDIRIZZO
UNITA'
DECINE
OPERANDO
REGISTRO

FIGURA 6·5

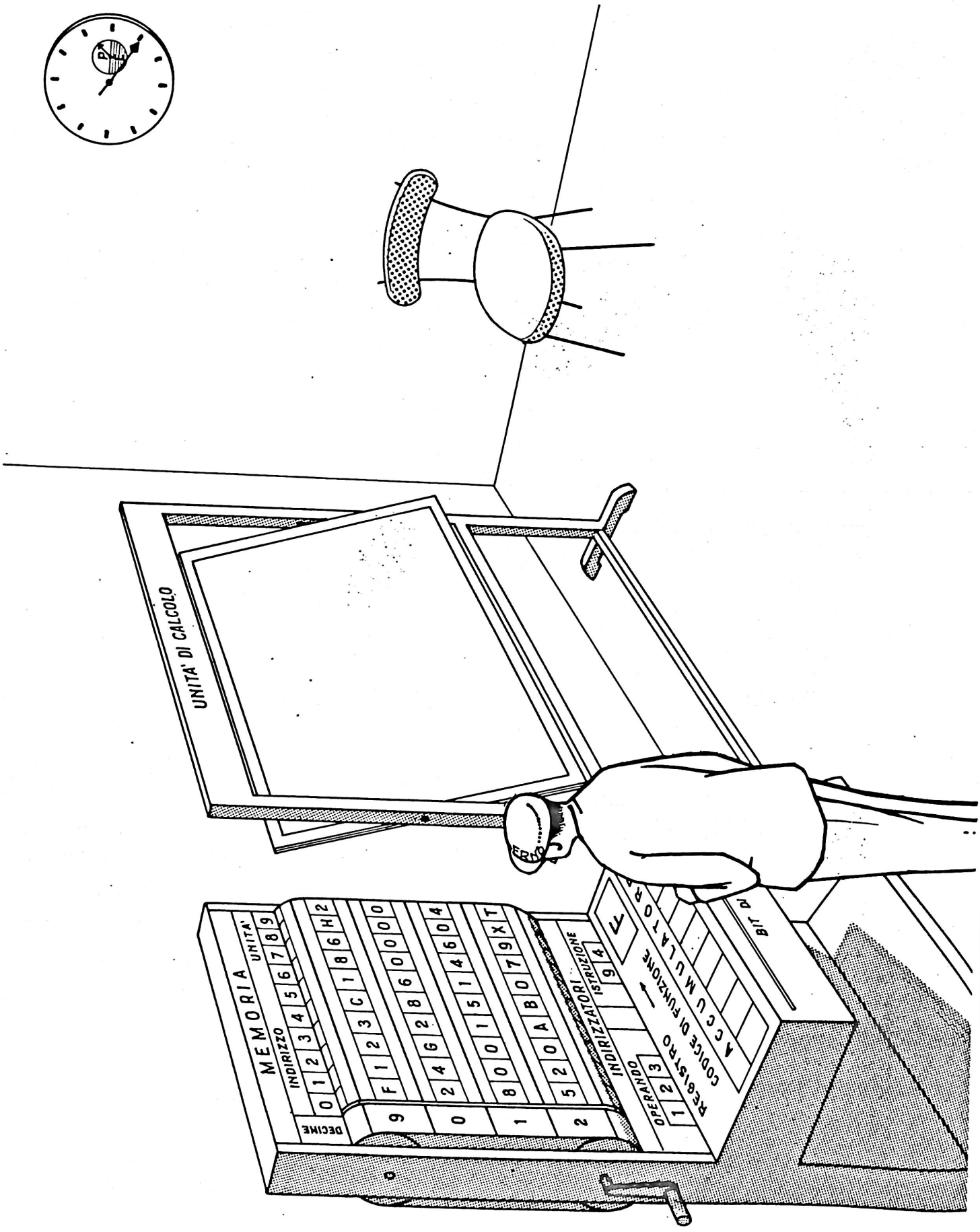

UNITA' DI CALCOLO
MEMORIA
UNITA'
INDIRIZZO
DECINE
INDIRIZZATORI
ISTRUZIONE
OPERANDO
REGISTRO
CODICE DI FUNZIONE
ACCUMULATORE
BIT di

Figura 6.5.: il governo interpreta l'istruzione. A questo punto l'orologio segna: fase preparatoria, periodi di cifra 4.

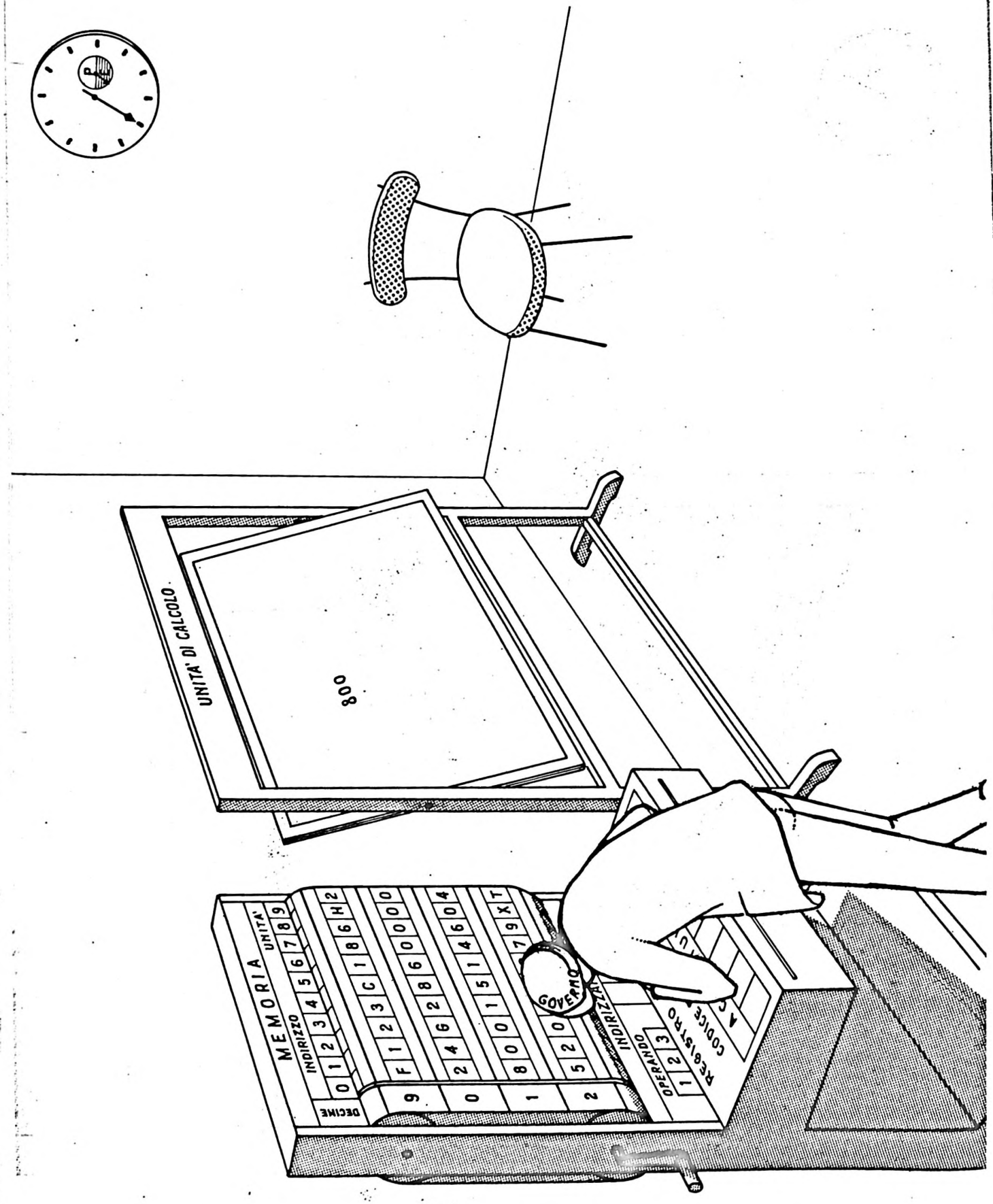
I disegni delle fig. 6.4 ,6.5 hanno illustrato lo svolgimento della fase preparatoria dell'istruzione.

In fig. 6.6 il governo trascrive nell'unità di calcolo l'operando individuato in memoria a partire dall'indirizzo 12 per lunghezza 3 caratteri, (800) quindi

FIGURA 6•6

UNITA' DI CALCOLO
MEMORIA
INDIRIZZO
UNITA'
DECINE
INDIRIZZATORI
ISTRUZIONE
OPERANDO
REGISTRO
CODICE DI FUNZIONE
ACCUMULATORE
BIT DI FERMO

FIGURA 6·7


UNITA' DI CALCOLO
800
MEMORIA
INDIRIZZO
UNITA'
DECINE
GOVERNO
INDIRIZZI
OPERANDO
REGISTRO
CODICE

come indicato in fig. 6.7, riporta la stessa quantità nell'accumulatore, il bit di fermo si posiziona sotto l'ultimo carattere trasferito, cioè sotto l'8.

L'orologio segna : fase esecutiva, periodi di cifra 7, vale a dire 4 della precedente fase preparatoria più 3 per i caratteri elaborati.

L'indirizzatore, fin dal termine della fase preparatoria dell'istruzione precedente, indica che la nuova istruzione da eseguire parte dall'indirizzo 94.

In fig. 6.8 il governo legge ed interpreta l'istruzione. L'orologio segna fase preparatoria e conta altri 4 periodi di cifra, per cui si arriva ad 11.

FIGURA 6·8

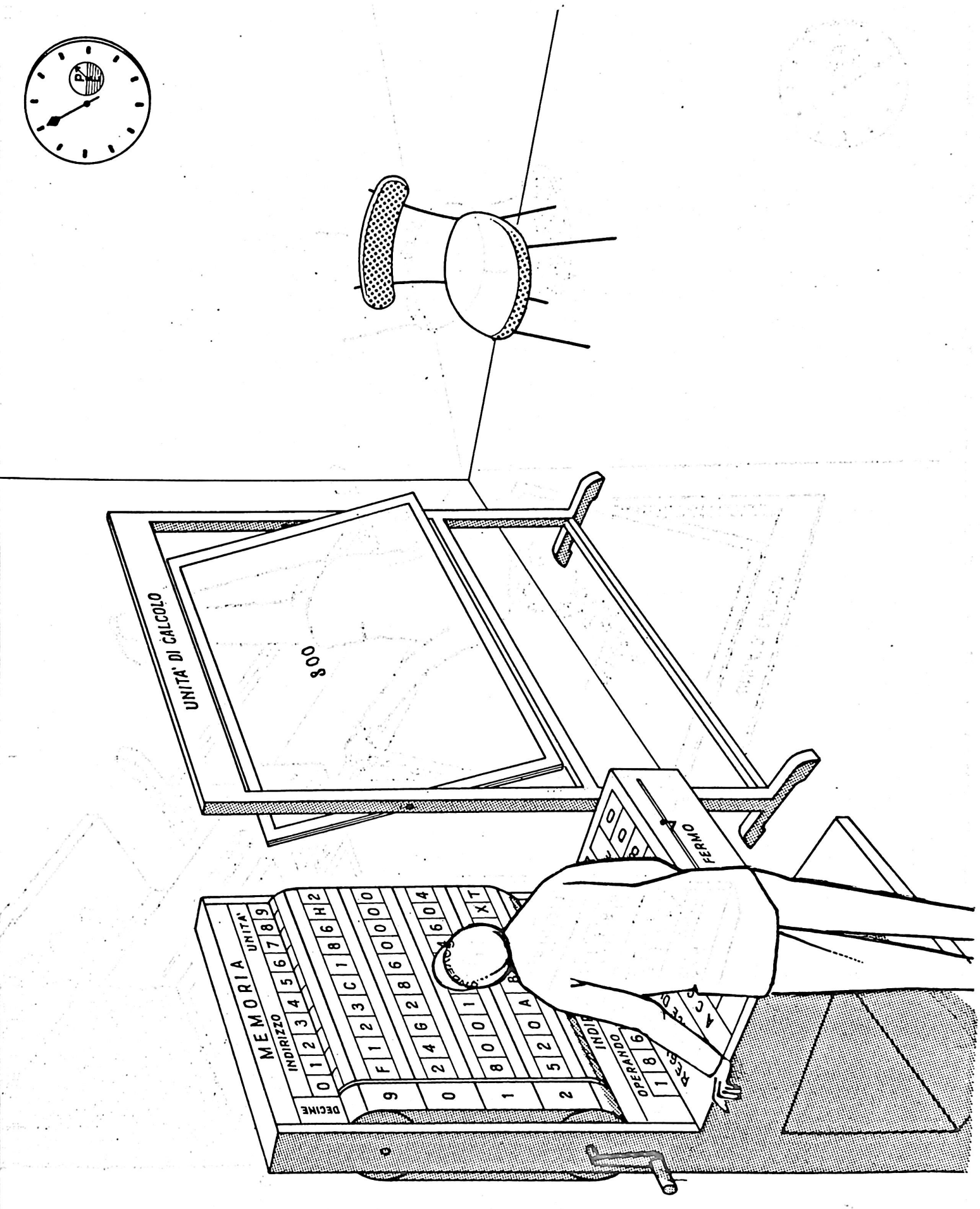

UNITA' DI CALCOLO
800
MEMORIA
INDIRIZZO
DECINE
UNITA'
OPERANDO
FERMO

FIGURA 6·9

UNITA' DI CALCOLO
800
151.460 +
MEMORIA
UNITA'
INDIRIZZO
DECINE
INDIRIZZATORI
ISTRUZIONE
OPERANDO
REGISTRO CODICE DI FUNZIONE
ACCUMULATORE
BIT DI FERMO

La fig. 6.7 mostra in fase esecutiva la somma dei due operandi 800 e 151460 in Unità di calcolo.

Il totale 152260, come ci illustra la fig. 6.10, viene quindi portato in accumulatore, con spostamento del bit di fermo sotto l'ultima cifra trasferita, cioè sotto l'1.

L'orologio indica fase esecutiva e conta altri 6 periodi di cifra per un totale di 17.

FIGURA 6 • 10

UNITA' DI CALCOLO
800
151.460 +
152.260
MEMORIA
INDIRIZZO
UNITA'
DECINE
GOVERNO
INDIRIZZATORI
ISTRUZIONE
OPERANDO
REGISTRO
CODICE DI FUNZIONE
ACCUM
BIT DI FE

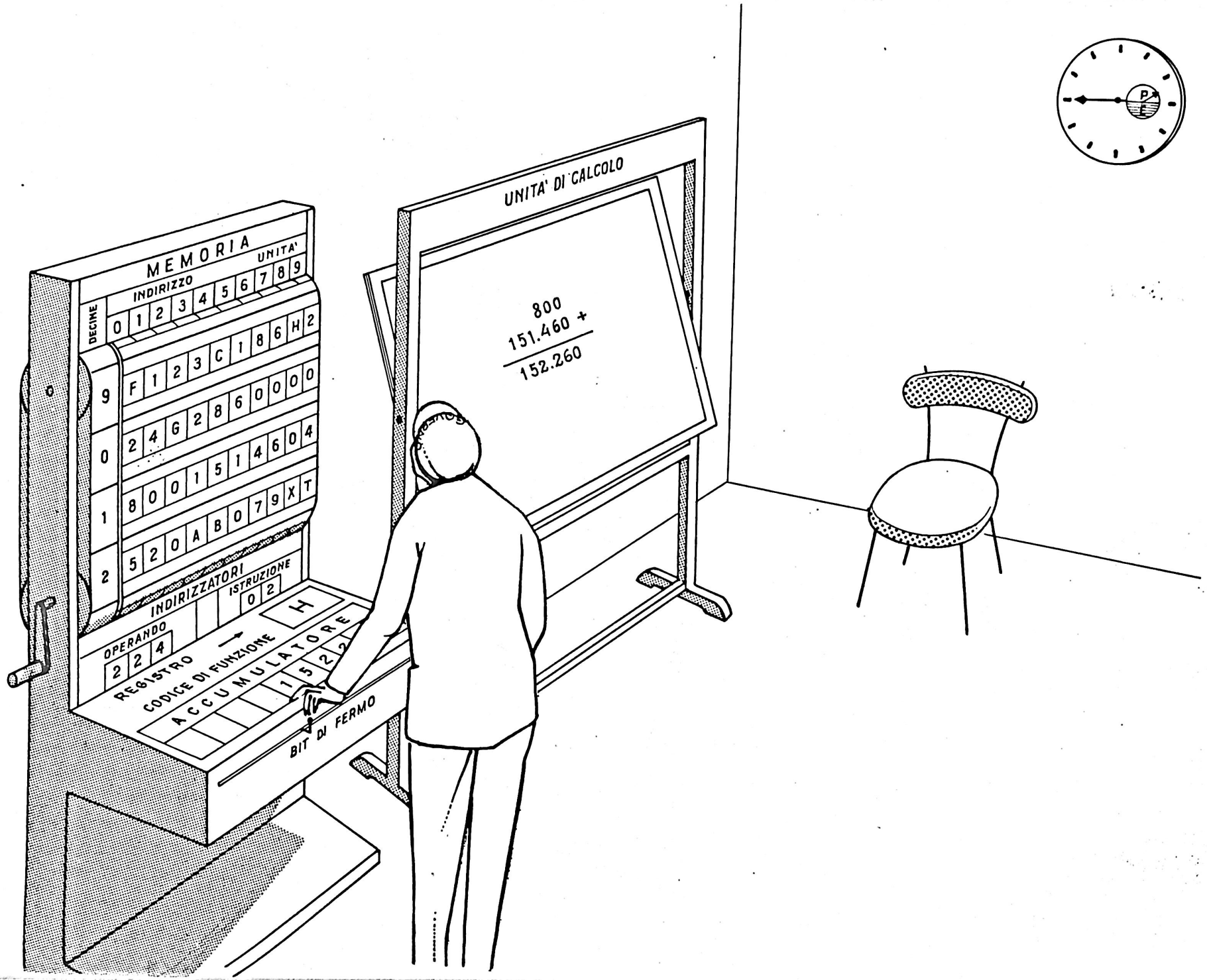

MEMORIA
INDIRIZZO
UNITA'
DECINE
0 1 2 3 4 5 6 7 8 9
9 F 1 2 3 C 1 8 6 H 2
0 2 4 G 2 8 6 0 0 0 0
1 8 0 0 1 5 1 4 6 0 4
2 5 2 0 A B 0 7 9 X T
INDIRIZZATORI
ISTRUZIONE
0 2
H
OPERANDO
2 2 4
REGISTRO
CODICE DI FUNZIONE
ACCUMULATORE
1 5 2
BIT DI FERMO
UNITA' DI CALCOLO
800
151.460 +
152.260

FIGURA 6·11

Finita la seconda istruzione l'indirizzatore indica al Governo che la nuova istruzione è all'indirizzo 98.

Uno dopo l'altro i caratteri dell'istruzione vengono prelevati e trascritti nel registro codice di funzione e nel l'indirizzatore d'operando (fig. 6.11).

L'orologio indica fase preparatoria e segna 4 periodi di cifra in più, cioè 21, l'indirizzatore istruzioni segna che la prossima sarà a 02.

La fig. 6.12 ci mostra lo svolgimento della fase esecutiva. All'accumulatore viene sottratto il termine di memoria a partire da 22 per 4 caratteri cioè 4520. Il risultato 147740 operato dall'unità di calcolo va in accumulatore.

Non c'è spostamento del bit di fermo, perchè il numero di cifre del risultato è uguale a quello precedentemente contenuto in Accumulatore.

L'orologio segna fase esecutiva con 4 periodi di cifra in più, per un totale di 25.

FIGURA 6 · 12

UNITA' DI CALCOLO
152.260
4.520 -
147.740
MEMORIA
UNITA'
INDIRIZZO
DECINE
GOVERNO
INDIRIZZATORI
ISTRUZIONI
OPERANDO
REGISTRO
CODICE DI FUNZIONE
ACCU
BIT

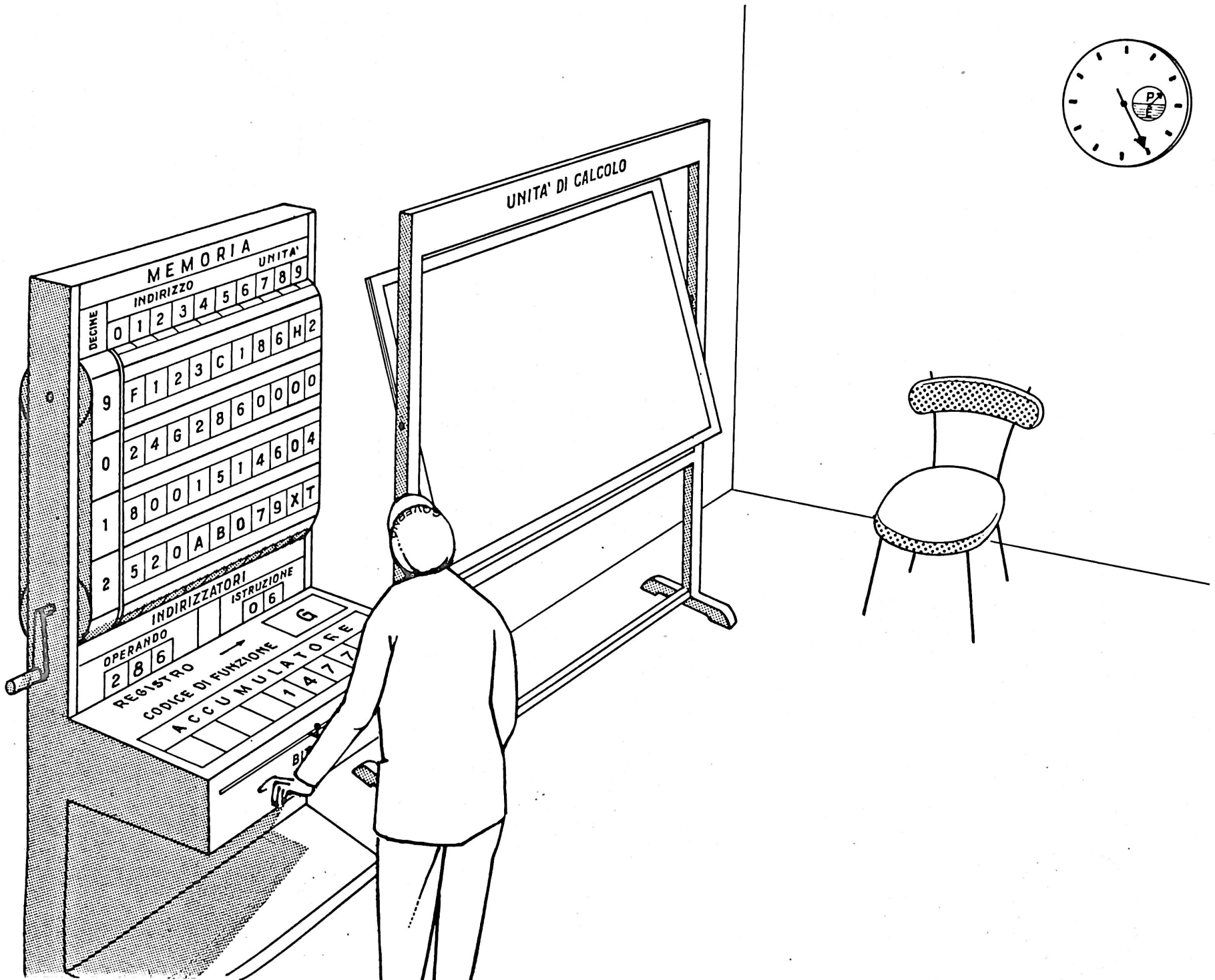

MEMORIA
INDIRIZZO
UNITA'
DECINE
0 1 2 3 4 5 6 7 8 9
9 F 1 2 3 C 1 8 6 H 2
0 2 4 G 2 8 6 0 0 0 0
1 8 0 0 1 5 1 4 6 0 4
2 5 2 0 A B 0 7 9 X T
INDIRIZZATORI
ISTRUZIONE
0 6
G
OPERANDO
2 8 6
REGISTRO
CODICE DI FUNZIONE
ACCUMULATORE
1 4 7
UNITA' DI CALCOLO

FIGURA 6·13

La nuova istruzione è indicata all'indirizzo 02.

Il Governo la legge e la interpreta: si tratterà di un trasferimento dell'Accumulatore in memoria. L'orologio segna fase preparatoria e periodi 29. (fig. 6.13).

Nella fase esecutiva l'operando di accumulatore viene trasferito in memoria attraverso l'unità di calcolo all'indirizzo indicato dall'istruzione per 6 caratteri. L'orologio segna fase esecutiva con altri 6 periodi di cifra per un totale di 35. (fig. 6. 14).

FIGURA 6·14

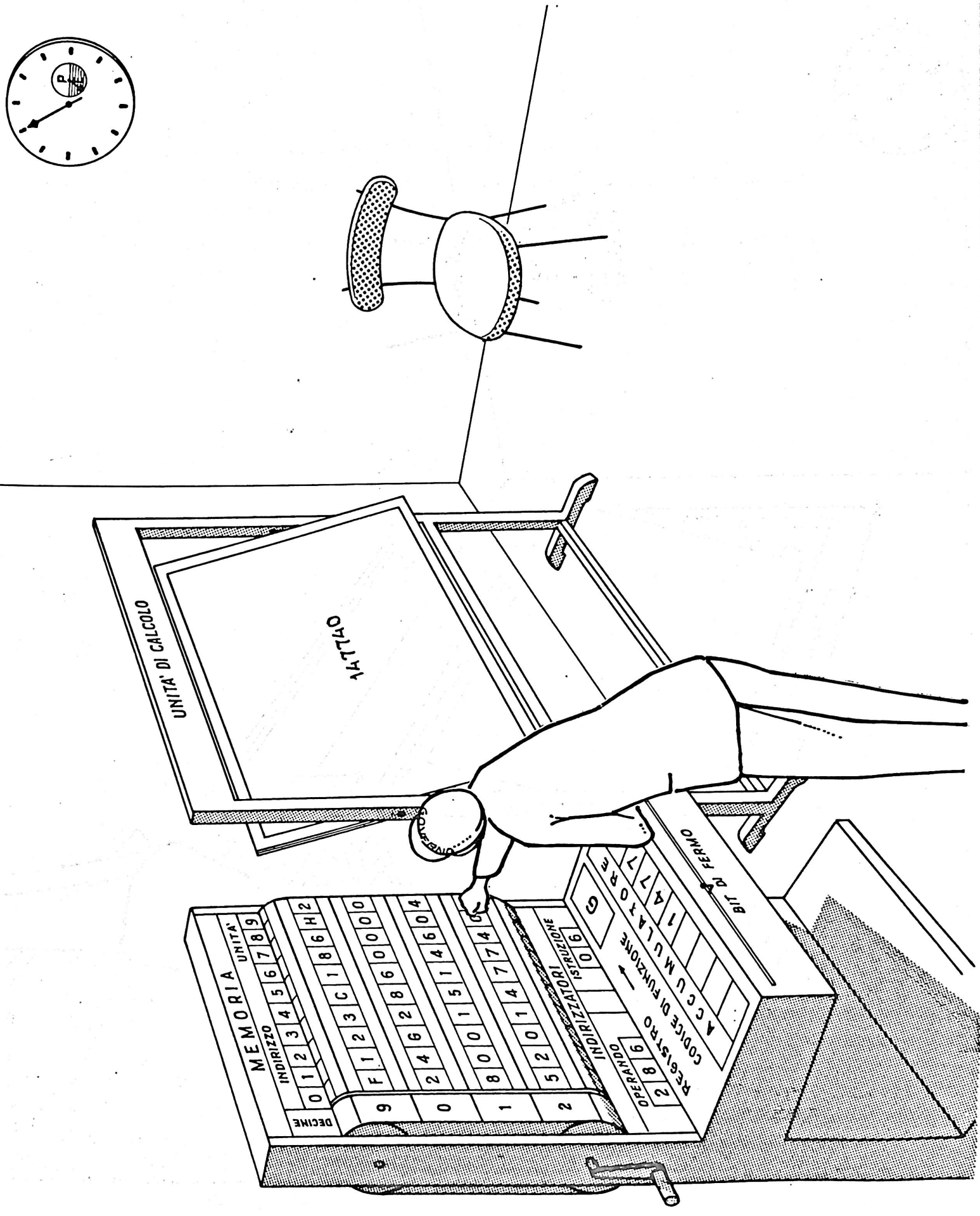
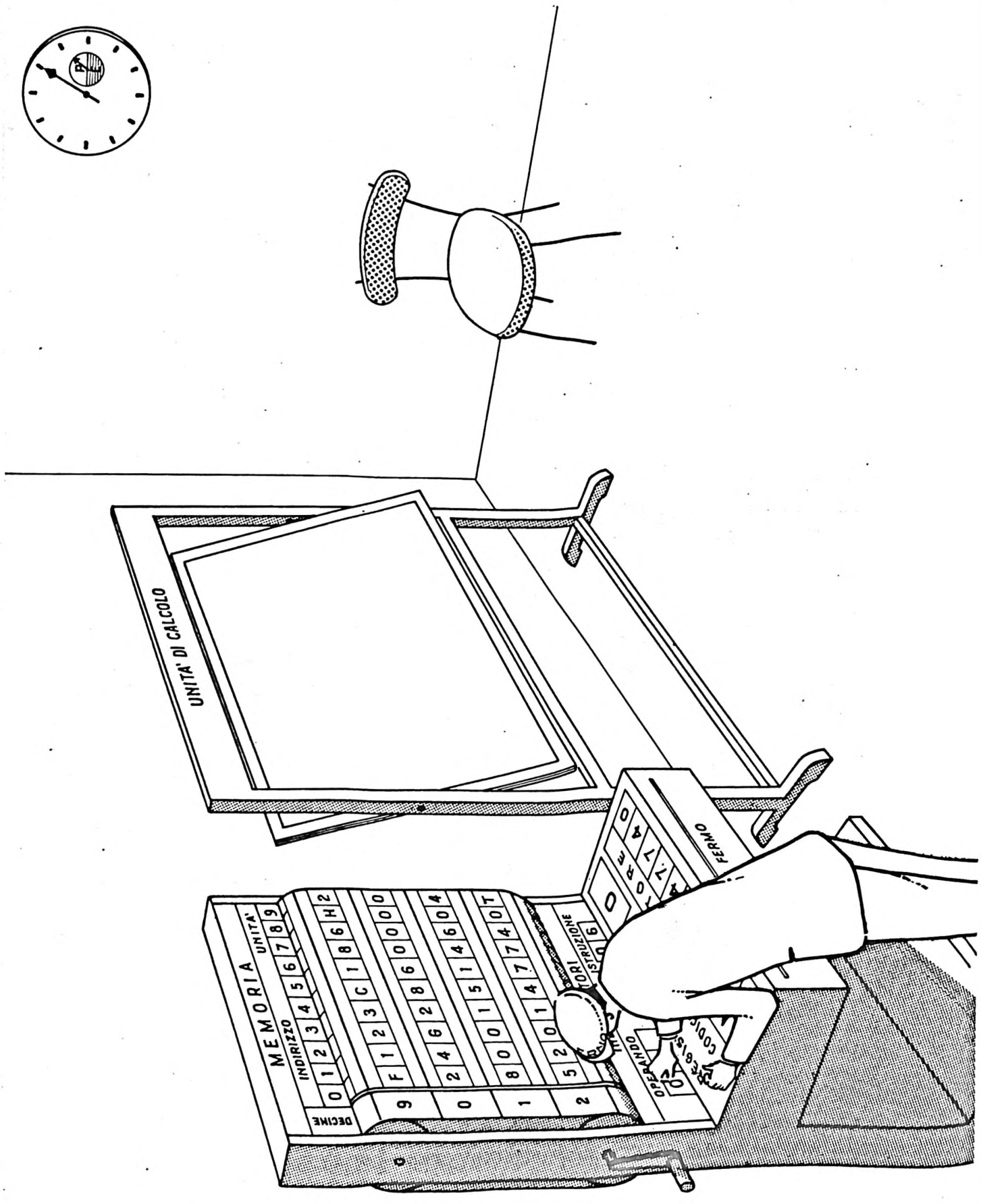

UNITA' DI CALCOLO
1A7740
BIT DI FERMO
REGISTRO
OPERANDO
INDIRIZZATORI
ISTRUZIONE
CODICE DI FUNZIONE
ACCUMULATORE
MEMORIA
INDIRIZZO
UNITA'
DECINE

FIGURA 6 • 15


UNITA' DI CALCOLO
MEMORIA
INDIRIZZO
UNITA'
DECINE
ISTRUZIONE
OPERANDO
FERMO

La nuova istruzione è all'indirizzo 06.

E' il comando di stop, in base al quale il Governo cessa di operare.

L'orologio si ferma sui 39 periodi di cifra. (figura 6.15 e 6.16).

FIGURA 6·16

UNITA' DI CALCOLO
MEMORIA
INDIRIZZO
DECIME
UNITA'
INDIRIZZATORI
ISTRUZIONE
OPERANDO
REGISTRO
CODICE DI FUNZIONE
ACCUMULATORE
BIT DI FERMO

P A R T E T E R Z A

LA PROGRAMMAZIONE

I LA PROGRAMMAZIONE

1.1 Funzione generale di un Elaboratore

Accennando alla funzione degli Elaboratori (parte 2^, 1.2), si è detto che essi sono in grado di provvedere alla risoluzione di qualunque tipo di problema, sottolineandone l'aspetto "universale". E' questa una caratteristica "a priori" degli Elaboratori, nel senso che potenzialmente la loro logica consente di soddisfare qualsiasi esigenza di elaborazione.

Si dice pure che gli Elaboratori sono degli automatismi completamente "aperti", intendendo con ciò sottolineare che seno accessibili a problemi di diversa natura e portata.

L'apporto degli Elaboratori nel campo del calcolo e dell'automazione aziendale è stato decisivo per il concorso simultaneo di tre fattori di fondamentale importanza: velocità, tempestività, e levata possibilità di decisioni logiche.

La velocità. Rispetto ai vecchi e tradizionali sistemi, l'utilizzazione di componenti elettronici ha realizzato quella notevole scioltezza, in base alla quale anche le più lunghe e complesse procedure di calcolo e di elaborazione appaiono facilmente eseguibili. Si pensi che il numero delle funzioni, che un Elaboratore può correttamente svolgere in un secondo, raggiunge il valore delle decine di migliaia!

Questo aspetto è tanto più significativo, se si considera che ha reso possibile sviluppare espressioni matematiche lunghe e laboriose, che diversamente avrebbero richiesto tanto tempo da sconsigliarne il calcolo a priori. Moltissime discipline, che prima esistevano in una forma soltanto embrionale, hanno avuto, un "exploit" prodigioso grazie all'impiego degli Elaboratori!

Lo stesso dicasi nel campo aziendale, dove innanzitutto sveltire una procedura significa diminuire l'incidenza dei costi amministrativi insieme ad una maggior precisione dei lavori e dei servizi.

I tradizionali Centri Meccanografici hanno assai presto dimostrato la loro insufficienza di fronte a problemi di vasta e cospicua mole.

Il potenziamento e la concentrazione industriale degli ultimi decenni ha visto accrescere il numero dei dipendenti delle aziende da poche migliaia a decine e persino centinaia di migliaia; il problema delle paghe e degli stipendi e il corrispondente aggiornamento dell'archivio anagrafico sono diventati dei pesanti problemi amministrativi.

Così l'allargarsi dei mercati di vendita, insieme alla produzione di una più vasta gamma di prodotti ha intensificato il volume degli ordini e del fatturato, dei mezzi di regolamento finanziario, delle rilevazioni statistiche ed extracontabili ad un punto tale, che solo la grande velocità degli elaboratori elettronici ha reso possibile il disbrigo di tutte queste pratiche.

La tempestività. Essa consente un adeguato intervento dell'Elaboratore nei problemi cosidetti di "real time", in contrapposto ai problemi di "delay time".

Il concetto di "real time" (tempo reale) è rapportato alla necessità che la rilevazione e quindi la risoluzione di particolari fenomeni venga effettuata al tempo stesso della loro manifestazione, onde consentire provvedimenti opportuni, che molto spesso sono emanazione diretta dei calcoli dell'elaboratore.

Controllare istante per istante l'andamento della traiettoria di un missile e segnalare le eventuali storture; coordinare una rete di semafori in base al volume del traffico; seguire lo svolgimento di una vasta catena di distribuzione per rilevare alcune anomalie nelle vendite, la cui comunicazione tempestiva e immediata è di vitale importanza per l'economia dell'azienda... la risoluzione di questi problemi è prerogativa esclusiva degli Elaboratori elettronici.

Elevate possibilità di decisioni logiche. Ci si riferisce alla facoltà che hanno gli Elaboratori (a maggior ragione quelli di

grande potenza) di prendere delle "decisioni", che sconfinano dal la sfera puramente esecutiva dei calcoli, e lasciano alla macchina il compito di "gestire" un determinato fenomeno.

L'evasione automatica degli ordini di un'azienda, la gestione di una polizza assicurativa, il controllo di una rete di traffico stradale, la determinazione di parametri di produzione, permettono di vedere nell'utilizzazione di un elaboratore non soltanto un semplice strumento, atto a sveltire enormemente le procedure, bensì un valido ed efficace contributo a problemi di direzione nel campo aziendale e a nuovi orientamenti nel campo tecnico e scientifico.

E' questo un fattore molto importante, che non deve essere assolutamente trascurato nella valutazione dell'apporto degli Elaboratori nel campo dell'automazione e della cibernetica.

Tutto ciò non deve indurre a pensare che l'Elaboratore dispensi l'uomo da ogni sforzo intellettuale.

La macchina è in effetti un semplice "automa", capace di effettuare a grande velocità tantissime operazioni, ma in modo particolare solo quelle che l'uomo le avrà comandato.

1.2 Le funzioni logiche e operative di un Elaboratore sono elementari

L'elaboratore, pur essendo un automatismo di grado assai elevato, è sempre una macchina, vale a dire uno strumento creato dall'uomo e per tanto dotato di un numero limitato di funzioni.

Ciò non è affatto in contraddizione con quanto si è detto precedentemente, anche se può sembrarlo a prima vista.

Le finalità che un elaboratore ci consente di perseguire sono praticamente illimitate, limitate invece ed elementari sono le modalità funzionali, mediante le quali provvede al conseguimento di tali finalità.

Limitate, perchè, come si è detto, è una macchina; elementari, perchè più semplici sono le funzioni di base maggiore (anzi elevatissimo) è il numero delle funzioni complesse che si possono ottenere dalla loro combinazione.

E' un po' la storia della riga e del compasso, due semplici strumenti, coi quali è possibile disegnare tantissime figure geometriche!

Questo non deve stupire nel campo del calcolo e delle elaborazioni, se si pensa che tutte le operazioni matematiche, anche le più complesse, possono ricondursi alla sola addizione e sottrazione. Dal punto di vista della semplice funzione operativa, un elaboratore che fosse dotato della sola addizione e sottrazione, purchè velocissimo, non precluderebbe nessuna via ai calcoli anche più complicati. Lo stesso dicasi per ordinare una serie di numeri da minore a maggiore o viceversa, è sufficiente che l'elaboratore sia dotato di una semplice funzione di confronto e di trasferimento.

L'esperienza dimostra che tutte le funzioni complesse possono essere scomposte in funzioni più semplici ed elementari, dalle quali, risalendo in senso opposto il cammino percorso, è consentito ottenere le funzioni stesse.

Come da un centinaio di elementi (idrogeno, ossigeno, sodio, potassio, ferro, ecc.) del sistema periodico si è giunti per via naturale o artificiale a comporre tutte le sostanze materiali esistenti, così con poche funzioni elementari un elaboratore arriva a soddisfare qualunque tipo di risoluzione.

1.3 Necessità di precisione

Perchè un calcolatore possa risolvere un problema, occorre che il problema esista realmente e che sia stato definito con precisione.

E' questa una condizione che non deve essere dimenticata. La macchina infatti, come qualsiasi automatismo esistente, è perfettamente stupida, ragiona, opera ed elabora solo e unicamente nel modo che gli dettiamo noi. Le operazioni da comandare alla macchina

debbono pertanto essere ben conosciute da principio e ad un livello nient'affatto vago e generico, ma assai preciso e soprattutto elementare. La precisione nella formulazione dei problemi è una condizione indispensabile alla loro corretta risoluzione.

Non si può infatti dire alla macchina semplicemente:

"Calcola un determinante"

"Calcola la criticità di un reattore"

"Calcola un ponte sospeso a tre luci"

"Provvedi ad evadere gli ordini di una azienda"

"Aggiorna i conti correnti di una Banca"

"Calcolami queste fatture"

"Traduci in lingua francese".

Per le applicazioni e le risoluzioni di tipo scientifico, si dovranno definire i problemi in termini di sistemi di equazioni, di formule da calcolare. Le stesse funzioni matematiche a loro volta dovranno essere poste in termini elementari, i soli comprensibili all'elaboratore; mentre per le elaborazioni aziendali, si dovrà redigere un processo di ragionamento logico assai dettagliato, comparazione di codici, letture di tavole, confronti di quantità ecc.

1.4 Convenienza del problema da risolvere: la sua natura ciclica.

Ci sembra di sentirci domandare: "Ma se per risolvere un determinato problema dobbiamo procedere ad una sua disamina così minuta, precisa e particolareggiata, allora tanto vale che facciamo a meno del calcolatore?"

Osservazione esatta dalla quale consegue che l'impiego della macchina è redditizio, solo quando il problema deve essere ripetuto un gran numero di volte. La fondamentale caratteristica che deve possedere un problema per essere utilmente trattato da un elaboratore è quella di essere di natura "ciclica".

A questo proposito esistono due tipi di ciclicità per spiegare i quali consideriamo un esempio: la compilazione dei bollettini paga per gli operai di un'azienda.

1) La macchina eseguirà lo stesso tipo di operazioni, finchè vi saranno bollettini paga da calcolare (vale a dire tanti quanti sono gli operai).

2) La compilazione dei bollettini paga sarà effettuata con una certa periodicità nel tempo, per esempio ogni settimana.

Anche molti problemi scientifici hanno natura ciclica. Pur non avendo una ciclicità del genere delle paghe, che debbono essere calcolate tutte le settimane, non manca l'aspetto ciclico nel tempo.

Per esempio il calcolo della traiettoria si ripeterà ogni volta che si torni a presentare la necessità della traiettoria di un nuovo missile. Nei problemi scientifici si trova un altro aspetto ciclico, così ad esempio nel calcolo della traiettoria, dove si ripetono le stesse operazioni per la determinazione delle coordinate di ogni punto.

Questo aspetto è tipico di tutte le macchine automatiche a maggior ragione per quelle più "aperte". Si pensi un momento ad una pressa stampa-pezzi. Essa funziona a punzone e matrice, componenti essenziali di uno stampo. Evidentemente il meccanico di precisione addetto alla costruzione del punzone e della matrice, deve fare un lavoro di precisione e di cesello di gran lunga superiore a quello che occorrerebbe per costruire un "solo pezzo". Il lavoro dello specialista è giustificato solo in quanto vi sia la necessità di costruire una grandissima quantità di pezzi, necessità che si può ripetere senz'altro anche nel tempo.

1.5 Definizione di programma

La sequenza di operazioni, incorporate nella logica della macchina, utilizzate per lo svolgimento di una determinata linea risolutiva, dicesi "programma".

Il concetto di programma presuppone quindi tre nozioni fondamentali:

1) l'esistenza di un problema,

2) l'esistenza di una logica funzionale di macchina,

3) uno svolgimento sequenziale.

Più propriamente scrivere un programma significa adottare la logica della macchina alle esigenze operative di un determinato problema, secondo uno svolgimento sequenziale.

Le esigenze operative del problema scaturiscono dalla logica risolutiva idonea al conseguimento dei risultati voluti e sono derivazione diretta della mente umana.

La logica funzionale di macchina è l'insieme di operazioni elementari che questa è in grado di espletare, scelte dal programmatore in modo da ubbidire alle finalità del problema.

Infine l'andamento sequenziale richiama la modalità di svolgimento dinamico, intesa come successione nel tempo delle operazioni di cui si è detto.

1.6 Le tre fasi di risoluzione di un problema

Da tutto ciò consegue che, una volta stabilito di risolvere un problema con un Elaboratore elettronico, conviene passare attraverso una serie di tappe intermedie, che condurranno in ogni caso alla fine alla scrittura del programma.

Possiamo decomporre queste operazioni preliminari in differenti fasi di lavoro che chiameremo:

1) Analisi del problema

2) Stesura di una linea operativa funzionale

3) Stesura della linea operativa macchina (codifica).

Alle tre fasi corrispondono rispettivamente al lavoro di analisi, programmazione vera e propria, codificazione.

Queste tappe preparatorie hanno molta rilevanza, infatti questi compiti che possono sembrare pedanti e fastidiosi a prima vista devono essere eseguiti con la massima diligenza, dalla quale non consentono di prescindere nè una conoscenza perfetta della mac - china nè la grande esperienza e abilità dei tecnici di programma zione. Proprio in funzione di ciò, nascono tre distinte mansioni e specializzazioni del lavoro di programmazione, le quali,con riferimento ai tecnici chiamati ad espletarle, danno vita alle figure rispettivamente:

1) dell'analista ,

2) del programmatore ,

3) del codificatore.

Si fa rilevare che queste mansioni possono essere svolte anche da una stessa persona, specialmente per quanto riguarda quella di programmatore e di codificatore.

E S E R C I Z I

1) Cosa significa dire che un Elaboratore è universale?

2) Quali requisiti fondamentali hanno permesso ai calcolatori di offrire un decisivo contributo all'automazione? Perchè?

3) Cos'è un problema di real time?

4) Perchè le funzioni di un elaboratore sono elementari?

5) Come si spiega "l'universalità" di un elaboratore in rapporto alla limitatezza delle sue funzioni di base?

6) Quale requisito fondamentale deve possedere un problema per essere risolto utilmente da un Elaboratore?

7) Quali tipi di ciclicità conoscete?

8) Cos'è un programma, inteso in senso logico?

9) Attraverso quali fasi si perviene alla risoluzione di un problema?

II L'ANALISI

2.1 Fase di analisi

La fase di analisi suppone evidentemente l'esistenza del problema di automatizzare che consiste di norma in un lavoro o servizio, di natura scientifica, tecnica o amministrativa.

Molto spesso, specialmente per i problemi commerciali, l'automazione si spinge molto più avanti e si rivolge all'intera organizzazione dell'azienda. Per cui non esiste più "un lavoro", "un servizio", ma un complesso integrato di lavori e servizi.

Posto di fronte ad un determinato problema l'analista, inizia lo studio degli aspetti che lo caratterizzano, chiarisce con gli utenti i punti oscuri, collabora con essi al fine di dare una impostazione corretta ed univoca.

L'analisi di problemi scientifici richiederà indubbiamente una specifica competenza nella materia da studiare e da esaminare. Lo stesso dicasi per i problemi commerciali, per i quali in particolar modo occorre uno studio organico delle funzioni, delle informazioni, e dei supporti di introduzione e di estrazione.

Nello studio dei problemi aziendali l'analista dovrà considerare a fondo i vari profili della materia, studiando quella particolare struttura organizzativa, in rapporto alle finalità del lavoro.

Una volta completato lo studio, attraverso la più oculata e attenta disamina dei vari aspetti, egli dovrà cercare:

1) di prospettare le soluzioni in rapporto all'utilizzazione di un elaboratore elettronico,

2) di integrarle cogli altri lavori dell'azienda.

Il punto 1) sottolinea l'aspetto attivo e dinamico dell'Elaboratore, volto a proporre soluzioni nuove più confacenti a l'economia del sistema elettronico, che consentono di utilizzare la capacità operativa della macchina, nella maniera più proficua per l'azienda.

Se ad esempio c'era prima un centro meccanografico a schede perforate, che colle sue modeste capacità lavorava ad un certo livello di rendimento, si cercherà di modificare gli schemi impiegati, nel senso di migliorare il rendimento non soltanto per la maggiore velocità dell'Elaboratore, ma in funzione delle sue maggiori possibilità operative.

L'Elaboratore è dunque in grado di suggerire soluzioni nuove più dinamiche e più razionali.

Il punto 1) è strettamente connesso al punto 2), il quale va studiato sotto l'aspetto del "Sistema Integrato di Elaborazione" che gli americani chiamano "Integrata Data Processing".

In un'azienda commerciale i servizi automatizzati, non costituiscono mai dei compartimenti stagni. Gli studiosi d'economia d'azienda sottolineano la struttura unitaria della impresa, dimostrando che fra i diversi suoi componenti esistono strettissimi rapporti di connessione e di interdipendenza, che non devono essere assolutamente trascurati ai fini di una corretta rilevazione.

L'analisi di un servizio deve perciò essere integrata, tenendo presente l'organicità della struttura aziendale in correlazione con gli altri servizi. Attraverso questo lavoro emergono dei fattori che consentono :

a) l'eliminazione di elementi superflui,

b) l'inserimento di elementi nuovi che eventualmente si erano trascurati o non si erano potuti considerare.

A questo punto lo studio si completa colla stesura di un diagramma a blocchi , sufficiente a dare al problema una visione organica delle procedure.

2.2 Definizione dei supporti e delle fasi di lavoro.

Eseguito lo studio, l'analisi del problema si spinge a definire :

1) la natura dei supporti,

2) il tracciato delle informazioni di base,

3) il diagramma a blocchi per il flusso dei dati, riproducente l'andamento delle fasi di lavoro.

La natura dei supporti da utilizzare è in diretta correlazione con tutto lo studio. Tuttavia la scelta dovrà essere quanto mai oculata, tenendo presente che il collegamento in linea coll'Elaboratore di supporti lenti, pregiudica l'economia del Sistema.

Vi sarà dunque una generale tendenza a ridurre i supporti banda e scheda a tutto aumento dei nastri magnetici, assai più pratici e soprattutto assai più veloci. Non si dimentichi che un Elaboratore è tanto più efficiente quanto più lavora a nastri magnetici.

Il tracciato delle informazioni di base, sia d'entrata che di uscita, avviene su appositi moduli, dove si specificano tutte le caratteristiche qualitative e quantitative delle informazioni stesse, ivi compresi i tracciati per la stampa.

Le diverse fasi di cui si compone l'esecuzione del lavoro vengono descritte dall'analista in forma letteraria e sinteticamente rappresentate in un diagramma a blocchi per il flusso dei dati, chiamato "Data flow-chart", dove fra l'al

FIGURA 2-1

DIAGRAMMI A BLOCCHI PER I DATI

SIMBOLI DI DATI

- *Simbolo generico di dati*

SIMBOLI SPECIFICI

a) Documento singolo

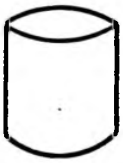
b) Documento stampato

c) Scheda perforata

d) Banda perforata

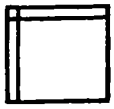
e) Nastro magnetico

f) Tamburo magnetico



g) Disco magnetico

h) Memoria a nuclei



i) Rappresentazione visiva

tro figurano la periodicità della fase e il suo tempo medio di svolgimento.

2.3 Diagramma a blocchi per il flusso dei dati.

Loro funzione è quella di descrivere il flusso dei dati attraverso il Sistema di Elaborazione. Questi diagrammi comprendono figure differenti riguardanti :

a) Linee di flusso, che indicano il trasferimento oppure il trasporto delle informazioni.

b) Simboli di dati che rappresentano i dati oppure i loro supporti.

c) simboli di operazione.

Questi diagrammi possono essere impiegati a due differenti livelli di analisi: uno generale che ancora non fa riferimento al sistema di Elaborazione dei dati, di cui si farà uso; un altro più dettagliato, che tiene conto del flusso di informazioni definito attraverso le unità di trattamento,che fanno parte del Sistema scelto per elaborare quella procedura. Ciascuna procedura per unità di trattamento definisce una "fase del lavoro".
Il primo caso è tipico di quei lavori d'analisi, a seguito dei quali si arriva a proporre lo strumento idoneo alla risoluzione dei problemi esaminati, che sarà l'Elaboratore tale o tal altro. Tuttavia anche quando il Sistema di Elaborazione è gia stato scelto, è buona norma in sede preliminare di analisi, fare un diagramma a blocchi delle procedure che serve di base per la stesura successiva del diagramma più dettagliato e particolareggiato, schema definitivo di tutto il lavoro d'analisi.
I paragrafi successivi descrivono i simboli da impiegare per la stesura dei diagrammi di flusso dei dati.

2.4 Simboli di dati.

Lo scopo principale di questi simboli è di definire l'esistenza dei dati e anche del mezzo attraverso i quali vengono forniti. Essi sono riportati in figura 2 - 1, alla quale è superfluo aggiungere commenti. Ci limitiamo a precisare che nell'interno di ciascun simbolo è bene inserire in forma scritta le precisazioni ulteriori. Per esempio : se in un determinato processo figurano delle schede perforate e dei nastri magnetici, indicare di quali schede si tratta e di quali nastri. Lo stesso vale per la memoria a nuclei e a tamburo, dove senz'altro sarà utile precisare il numero di posizioni.

2.5 Simboli di operazione.

Loro funzione è quella di indicare il tipo di operazione in base alla quale i dati vengono elaborati. Linee di flusso entranti e linee di flusso uscenti distinguono i dati di partenza da quelli successivi di arrivo. In aggiunta ai simboli di operazione si possono dare precisazioni circa l'unità tecnologica usata per l'operazione.

La figura 2.2 illustra i vari simboli. Diamo qui di seguito un breve commento, per quanto riguarda i simboli specifici di operazione.

Il Simbolo di inserimento : riguarda la formazione di una sola sequenza ordinata, a partire da due o più sequenze ordinate secondo un codice comune.
Il Simbolo di selezione individua l'operazione mediante la quale da un singolo complesso di informazioni si passa a due o più complessi, ognuno dei quali soddisfa a un criterio particolare.

Simbolo di inserimento e di selezione. In un primo tempo per mezzo di un inserimento si ottiene una sola sequenza or

FIGURA 2-2

DIAGRAMMI A BLOCCHI PER I DATI

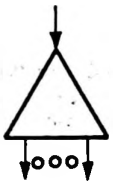
SIMBOLI DI OPERAZIONE

- *Simbolo generico di operazione*

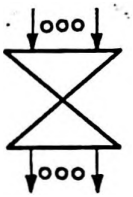
SIMBOLI SPECIFICI

a) Inserimento

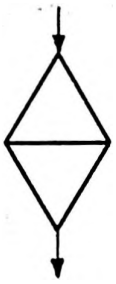
b) Selezione



c) Inserimento e selezione



d) Ordinamento



e) Intervento manuale

f) Operazioni manuali

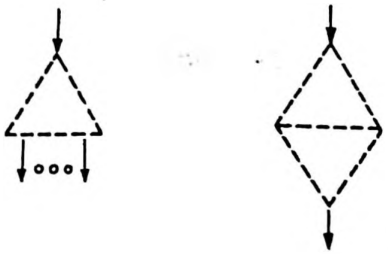


FIGURA 2-3. III

DIAGRAMMI A BLOCCHI PER I DATI

a) LINEE DI FLUSSO

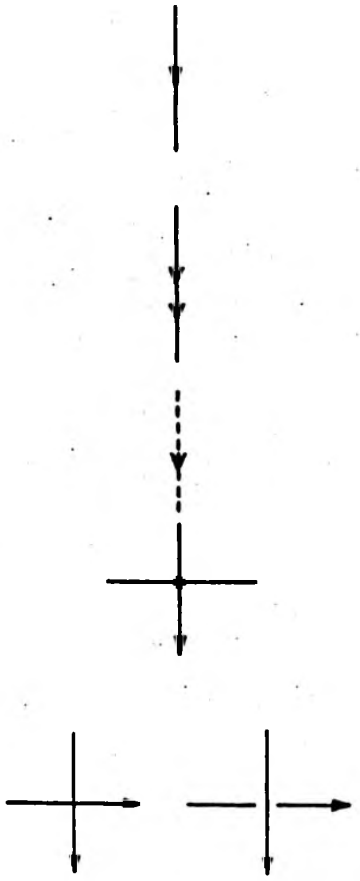
Trasferimento di dati senza trasporto del supporto relativo

Trasporto del supporto fisico dei dati

Trasferimento o trasporto ritardato :
interessa il collegamento con una fase di elaborazione di ordine diverso da quello normale.

Punto di confluenza

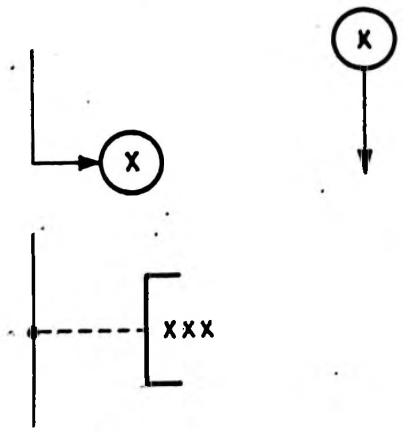
Linee senza intersezione



b) SIMBOLI AUSILIARI

Connettori

Commento



dinata, quindi secondo particolari modalità si estraggono da questa altre sequenze ordinate, normalmente differenti da quelle d'origine.

Il Simbolo d'ordinamento definisce il procedimento secondo il quale da un complesso disordinato si passa ad una sequenza d'informazioni ordinata secondo un certo codice, detto chiave dell'ordinamento.

Il simbolo d'intervento manuale è usato per indicare l'introduzione o la trascrizione di dati fatta manualmente dall'operatore o per indicare segnalazioni speciali che devono essere perforate.

Il simbolo di operazione manuale riguarda un'operazione di tipo automatico, che però diversamente dal solito, viene eseguita a mano.

2.6 Linee di flusso.

Nei diagrammi di flusso. Tutti i trasferimenti sono indicati con una linea continua, nella quale figura, se necessario, una freccia semplice. La distinzione fra differenti tipi di linee di flusso, è rappresentata in figura 2.3 a.

2.7 Simboli ausiliari.

Connettori. Si può sostituire una linea di collegamento con un connettore, figura 2.3.5, nel quale l'indicazione x rappresenta il carattere o il gruppo di caratteri d'identificazione. I due segnali di connessione possono trovarsi sullo stesso foglio o su fogli diversi.

Commento. Esso può essere collegato ad una linea oppure ad un altro simbolo (figura 2.3.5); xxx rappresentano una o più frasi di chiarificazione e talvolta un riferimento a qualcosa che non figura nel diagramma.

E S E R C I Z I

1) Cosa s'intende per complesso integrato di lavori e di servizi?

2) L'integrazione dei servizi quali vantaggi comporta?

3) Che cosa occorre definire in fase di analisi?

4) Perchè è importante definire la natura dei supporti?

5) Cos è il diagramma a blocchi per il flusso dei dati?

6) Disegnate i seguenti simboli:

   banda perforata

   nastro magnetico

   scheda perforata

   memoria a nuclei

   ordinamento

   operazione manuale d'introduzione

   operazione manuale di decisione

   connettore.

III LINEA OPERATIVA DI PROGRAMMA: IL DIAGRAMMA DI FLUSSO

3.1 Necessità di una linea operativa

Definite le fasi di un lavoro, occorre tracciare la linea operativa da seguire per svolgere convenientemente quanto è enunciato nella descrizione delle singole fasi.

L'analista in fondo compie un lavoro che fornisce i dati di partenza per la stesura della linea risolutiva, intesi sia come informazioni di entrata sia come "enunciato" del problema in oggetto. Da questi punti di partenza occorre iniziare il tracciato della linea risolutiva.

E' questa la seconda tappa del procedimento, attraverso il quale si perviene alla compilazione del programma oggetto; è il momento in cui si traccia il Diagramma di flusso (detto anche flow-chart), cioè si compone una serie di operazioni successive tali da consentire la via risolutiva.

Si tratterà evidentemente di operazioni intelleggibili alla macchina, proprie cioè della sua logica funzionale, di operazioni capaci di essere comandate, come si vedrà in seguito, mediante "istruzioni" o "subroutines".

3.2 L'importanza dell'enunciato d'analisi

La linea risolutiva, tracciata dal diagramma, è una impostazione che si dà ad un determinato problema (oppure ad una fase di un determinato problema), collegando all'enunciato le possibilità operative della macchina.

Risulta perciò evidente l'importanza dell'enunciato d'analisi. -

Può capitare che durante la stesura del Diagramma di flusso la maggior precisione imposta da questo nuovo linguaggio, sol

levi dei dubbi e trovi delle contraddizioni con l'enunciato di analisi. In tal caso, il programmatore farà presente all'analista le incongruenze incontrate.

Da quanto detto scaturiscono due punti di fondamentale importanza:

1) il programmatore deve necessariamente studiare a fondo la descrizione d'analisi, prendere nella dovuta considerazione i supporti e avere conoscenza precisa delle finalità del problema;

2) quando l'analista e il programmatore non sono la stessa persona, fra i due è necessario che si stabiliscano dei rapporti di stretta collaborazione.

3.3 Diagrammi a blocchi per il programma.

I diagrammi a blocchi si distinguono in due tipi :

Diagrammi a blocchi per i dati (Data flow-chart)
Diagrammi a blocchi per il programma (Program - flow-chart)

Mentre i primi, come si è visto al capitolo precedente, sviluppano il processo di elaborazione definendo il flusso delle singole fasi di lavoro, questi ultimi descrivono la linea risolutiva, rappresentando il flusso delle operazioni necessarie allo svolgimento di una fase determinata.

I diagrammi a blocchi non sono evidentemente un modo particolare di ragionare. Ma semplicemente un valido strumento per rappresentare un ragionamento.

Il ragionamento infatti preesiste al diagramma, il quale non è nient'altro che un linguaggio, idoneo ad esprimere chiaramente il concetto già presente nel ragionamento stesso. La logica su cui si fonda ogni ragionamento legato alla linea risolutiva di un problema è del tipo deduttivo - deterministico. Cioè è una logica che, partendo da determinate pre

FIGURA 3-1

DIAGRAMMI A BLOCCHI PER IL PROGRAMMA

SIMBOLI DI OPERAZIONE

- *Operazione generica*

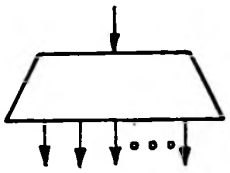
- *Sottoprogramma*

- *Entrata e Uscita*

- *Preparazione di una separazione*

SIMBOLI RELATIVI AL FLUSSO DELLE OPERAZIONI

- *Separazione*

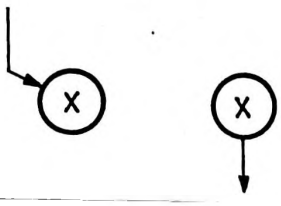


- *Confluenza*

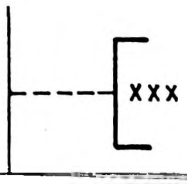
- *Procedura in parallelo*

SIMBOLI AUSILIARI

Connettori



Commento



messe, seguendo una data procedura, perviene a determinate conclusioni e a quelle sole.

E' un ragionamento di tipo matematicc, ma non è necessariamente un ragionamento matematico.

Tutto ciò è tanto più vero in quanto, questo ragionamento dovrà essere eseguito da una macchina, da un complesso che essendo stato progettato dall'uomo, non può avere una logica diversa da quella rigorosamente deterministica.

3.4 Simboli di operazione

Operazione generica : rappresenta ogni operazione che crea modifica trasferisce o cancella informazioni, oppure ogni altra operazione, per la quale non esistono simboli appositi. Il termine informazione è del tutto generale, esso comprende, dati, istruzioni, indicatori, ecc.

Sottoprogramma : questo simbolo si riferisce a una porzione di programma che, per i particolari scopi del caso specifico viene considerata come un'unica funzione. (vedi capitolo 4).

Operazione di entrata e uscita : viene usato ogni volta che si deve indicare un'operazione di introduzione o di estrazione dei dati. Esempio lettura registrazione di una scheda, di un blocco di nastro, ecc.

Preparazione di una separazione : serve per mettere in evidenza che una operazione determina parzialmente o completamente la scelta di una via in un simbolo di separazione. Tali sono gli aggiornamenti degli indicatori, dei contatori dai quali dipendono le successive decisioni, oppure i posizionamenti dei deviatori.
I simboli sono riportati in figura 3.1.

3.5 Simboli relativi al flusso delle operazioni.

Le operazioni sono collegate fra di loro da linee frecciate; quando il senso di procedura va dall'alto al basso la freccia può essere omessa.

Separazione : è costituita da una sola entrata e da due o più uscite, ogni volta però che si passa su questo simbolo viene usata soltanto una sola delle vie d'uscita. Ogni via d'uscita deve obbligatoriamente procedere in senso verticale, salvo a deviare più avanti.

Confluenza : è formato da più di un'entrata e da una sola uscita. Il simbolo consiste in un punto nero posto in corrispondenza dell'intersezione. Giungendo a questo simbolo per una qualsiasi delle entrate si prosegue sempre attraverso un unica uscita. Non esistono limitazioni a proposito dell'angolo che le linee formano incontrandosi. L'assenza del "punto nero" individua linee fra le quali non vi è nessuna connessione logica. In quest'ultimo caso però, onde evitare confusioni, è meglio spezzare una delle due linee in prossimità del punto d'incontro.

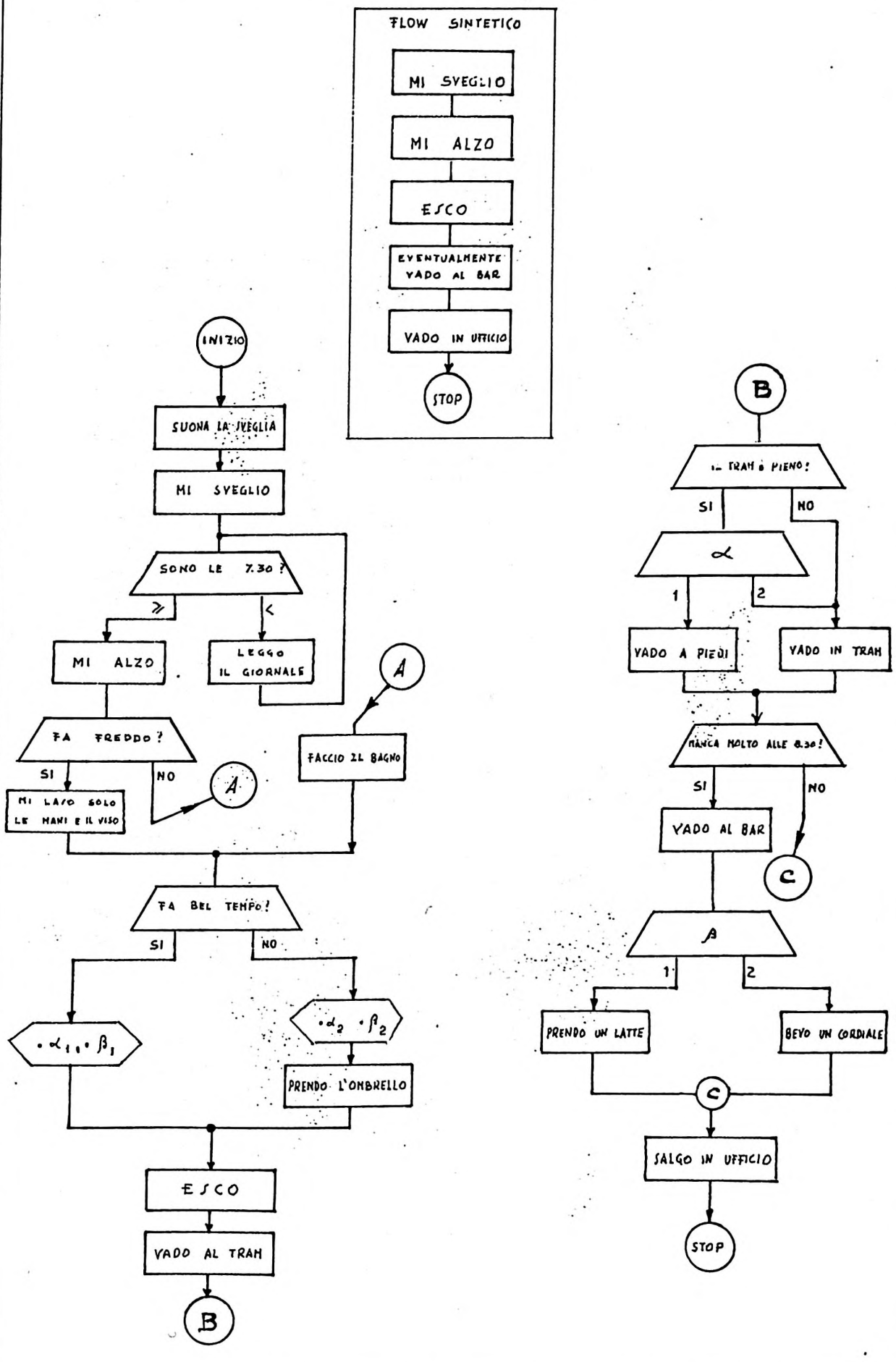
Procedura in parallelo : definisce un punto dal quale tutte le linee di uscita procedono in parallelo simultaneamente o in sequenza arbitraria.
I simboli sono riportati in figura 3.1.

3.6 Simboli ausiliari.

Connettori : sostituiscono una linea di collegamento, l'indicazione x rappresenta un carattere o un gruppo di caratteri, che servono per l'identificazione. I due segnali, iniziale e finale possono essere su uno stesso foglio o su fogli diversi, in tal caso occorrerà mettere una nota che consenta di rintracciare il foglio in cui si trova il connettore iniziale corrispondente. L'inizio e la fine di un programma sono in

FIGURA 3-2

ESEMPIO DI FLOW CHART - PROBLEMA: COME MI ALZO AL MATTINO



dicate con connettori particolari con indicazione rispettivamente di INIZIO e di STOP .

Commento : vale ad aggiungere in certi punti del programma spiegazioni più dettagliate. xxx è la dicitura esplicativa. I simboli sono riportati in figura 3.1.
Si veda l'esempio riassuntivo fig. 3-2: COME MI ALZO AL MATTINO.

3.7 Norme per la corretta applicazione di alcuni simboli.

Questo paragrafo intende chiarire il significato logico di certi simboli perchè non vi siano dubbi sulla loro esatta applicazione. La figura 3.3 riporta in diagrammi a blocchi gli esempi illustrativi.

Simbolo di operazione generica. Rappresenta una funzione operativa la cui natura può essere :

- aritmetica
- algebrica
- logica
- di trasferimento
- di ricerca.

Dopo quanto si è detto nella parte seconda al capitclo 4°su l'U.AL. non è il caso di illustrare ulteriormente queste operazioni.
Diciamo soltanto che le operazioni all'interno possono essere indicate in forma letteraria e in forma simbolica.
Se per esempio si deve moltiplicare la quantità per il prezzo di un prodotto al fine di ottenere l'importo lordo, si scriverà:

- Moltiplica quantità e prezzo,

oppure si potrà definire il prezzo con P, la quantità con Q quindi scrivere dentro al simbolo :

- P x Q

Ancora meglio, dato che il prezzo per la quantità darà luo-

go all'importo lordo IL, si scriverà:

$$P \times Q \longrightarrow IL,$$

che si legge "pi per Qu" "in" Ielle"

Facciamo un altro esempio: supponiamo di voler trasferire una quantità R in una variabile L: scriveremo

$$R \rightarrow L$$

che si legge "erre in elle".

Così pure se si vuole conferire ad una variabile X un valore noto e costante, si scriverà ad esempio:

$$5 \rightarrow x \; : \; 0 \rightarrow x \; : \; 1 \rightarrow x$$

vedi figura 3.3

La freccia sta a significare che il risultato dell'operazione di sinistra si trasferisce nella variabile di destra, che cambia pertanto il suo valore. Evidentemente se si tratta di un elaboratore a memoria "non distruttiva" le variabili di sinistra restano inalterate. Però se sia a sinistra che a destra figura una stessa variabile, alla fine dell'operazione in quella variabile ci si trova sempre il risultato. Per esempio, supponiamo di dover arrotondare un importo I.G.E. a 3 decimali per eccesso, scriveremo:

$$\text{I.G.E.} + 999 \longrightarrow \text{I.G.E.}$$

Quando il programmatore per ragioni di maggior chiarezza preferisce indicare l'operazione in maniera simbolica, é necessario integrare il diagramma con una legenda o glossario, dove i simboli usati vengono correttamente definiti. Se in un programma chiamo P il prezzo, Q la quantità, IL l'importo lordo ecc, prima dei diagrammi a blocchi scriverò:

GLOSSARIO

P = prezzo
Q = quantità
IL = importo lordo
ecc.

Simbolo di separazione. Si è detto che le operazioni hanno andamento sequenziale cioè si svolgono nel tempo una dopo l'altra. Vi saranno però dei punti in cui la sequenzialità si interrompe e si avranno delle separazioni.

La separazione può essere incondizionata, cioè può avvenire in ogni caso; oppure può essere collegata al verificarsi di una o più condizioni.

Il simbolo di separazione è usato solo in caso di smistamento soggetto a condizione. La condizione è esposta nel trapezio e le uscite legate ad essa sono specificate nei segmenti sottostanti, i quali ovviamente saranno pari al numero delle uscite possibili.
La condizione può esssere di due tipi differenti:

1) Condizione legata alle caratteristiche dei dati da elaborare, Es. se il versamento tale è superiore a 5000 prendi una certa via, altrimenti(cioè caso in cui sia uguale o inferiore) prendine un'altra.

   Normalmente in questo caso la condizione scaturisce da un confronto (o da una serie di confronti)indicatecon : : .

2) Condizione automatica di macchina. Essa si verifica automaticamente nell'Elaboratore, tali sono ad esempio i segnali di overflow (caso di risultato che supera una determinata dimensione) di fine sequenza, di fine flusso, di errore ecc.

   Per esempio se si è avuto segnale di fine flusso si debba eseguire un controllo, diversamente no.

Il simbolo di separazione viene usato comunemente in caso di decisione e di deviatore. Ciò dipende dal punto in cui avviene l'esame della condizione.
Nella decisione l'analisi della condizione è contestuale al simbolo di separazione e lo si è appena visto nei due esempi ripertati in figura. Il deviatore presuppone che l'esame della condizione sia stato fatto precedentemente col risultato di "posizienare" il deviatore in una delle sue vie.

Passando sul deviatore la via da seguire risulta così indicata dall'ultimo posizionamento. Il deviatore è contrassegnato da una lettera greca e le vie d'uscita vengono numerate progressivamente.

Preparazione di una separazione.

Queste simbolo individua tutte quelle operazioni che interamente e parzialmente preparano la separazione.
Quando la separazione è una decisione si fa riferimento ad aggiornamenti di indicatori, di contatori e variabili diverse le quali interverranno a modificare il risultato delle esame di una condizione.

Si supponga una condizione che si verifichi se K è uguale a 4, finchè K non è 4 si segue una via, dopo di chè se ne segue un'altra. L'aggiornamento del contatore K di 1, cioè il K+1—>K è di preparazione ad una separazione.

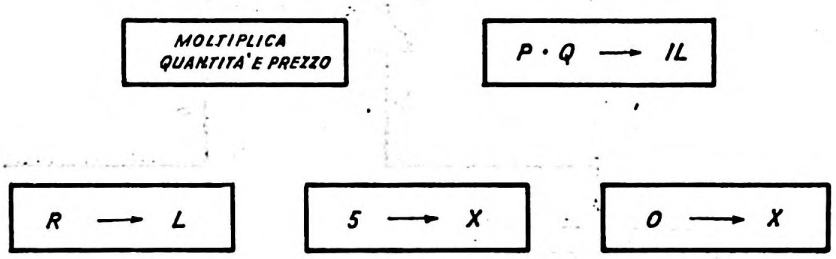
Se la separazione è un deviatore la preparazione consiste nei posizionamenti delle diverse vie.

Del simbolo di sottopregramma parleremo nel capitolo successive. (figura 4.1)

FIGURA 3-3

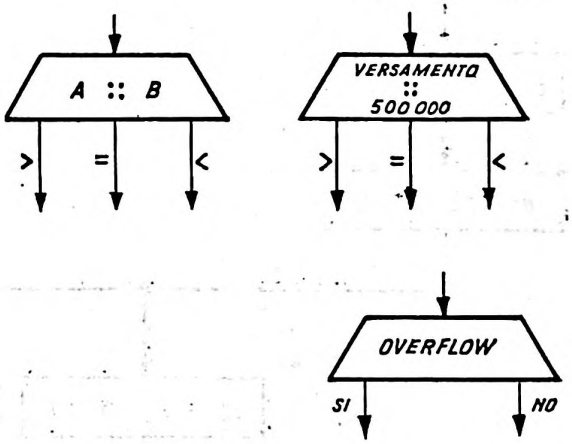
ESEMPI ILLUSTRATIVI

Operazione generica

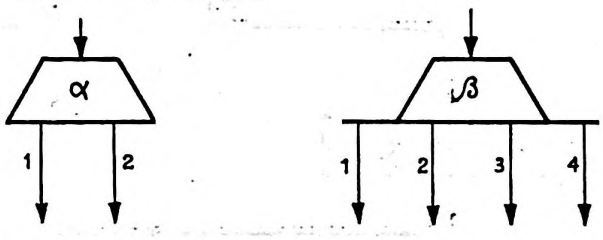


Separazione

Decisione

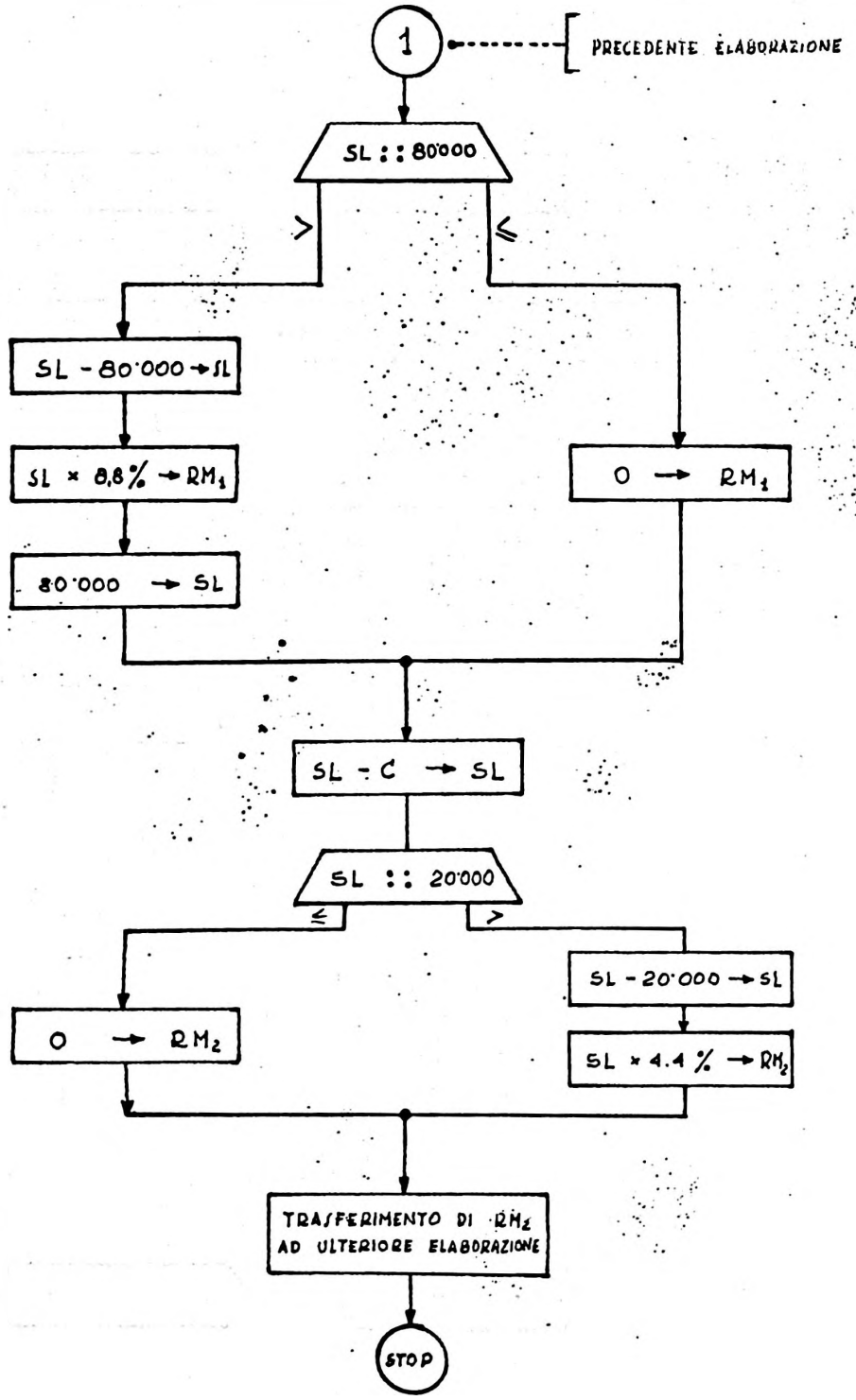


Deviatori



Preparazione di separazione

FIGURA 3-4


1
PRECEDENTE ELABORAZIONE
SL : : 80'000
>
≤
SL - 80'000 → SL
SL × 8,8 % → RM1
80'000 → SL
0 → RM1
SL - C → SL
SL : : 20'000
≤
>
0 → RM2
SL - 20'000 → SL
SL × 4,4 % → RM2
TRASFERIMENTO DI RM2 AD ULTERIORE ELABORAZIONE
STOP

3.8 Esempio di diagramma di flusso.

Si deve risolvere il problema relativo al calcolo della ricchezza mobile su uno stipendio.
Dall'analisi si rileva quanto segue:
la ricchezza mobile è il risultato della somma di due prodotti:
- primo, del 9,20% coll'importo della retribuzione eccedente le 80.000 lire di stipendio mensili.

- secondo, del 4,40% con la parte della retribuzione uguale o inferiore a 80.000, dopo aver detratto l'ammontare della franchigia fissa, pari a lire 20.000.
Se le competenze mensili risultassero d'importo uguale o inferiore a lire 20.000, nessun contributo deve essere versato a titolo di ricchezza mobile.

Poichè dentro ciascun blocco del diagramma intendiamo esprimere le operazioni in maniera simbolica e non descrittiva (la scelta è opzionale), chiamiamo con:
- SL lo stipendio lordo;
- C i contributi assicurativi e assistenziali, che per ipotesi supponiamo siano già stati calcolati precedentemente;
- RM_1 l'importo di ricchezza mobile, risultante dal prodotto del 9,20% con la somma eccedente le 80.000 lire;
- RM_2 l'importo di ricchezza mobile tassato al 4,40% (esso risulterà dal prodotto fra la parte uguale o inferiore a 80.000 lire, decurtata di 20.000, colla percentuale suddetta).
Una volta calcolati RM_1 e RM_2 si passerà ad una ulteriore elaborazione, prima di arrivare al blocco contrassegnato dallo STOP.
Sempre rifacendoci allo studio del problema, risulta evidente come, prima di ogni altra cosa, si debba effettuare il confronto fra l'ammontare dello stipendio lordo ed il valore di 80.000 lire.

Se lo stipendio risulta maggiore, gli togliamo 80.000, calcoliamo su questa differenza il 9,20%, la mandiamo in RM_1, quindi riportiamo lo stipendio lordo a 80.000 per effettuare il calcolo successivo di RM_2.

Se dal confronto lo stipendio è risultato uguale o inferiore, si mette zero in RM_1 e si passa al calcolo di RM_2. In ogni caso dunque le due linee operative si congiungono e provvedono a togliere 20.000 e C ad SL_{1-}

Quindi si dovrà confrontare con zero questa differenza, perchè se risulta maggiore si calcola il 4,40%, al contrario si porrà zero in RM_2 .

Dopo di che si passerà ad una elaborazione ulteriore per poi arrivare allo STOP.

Il diagramma di flusso è riportato in figura 3.4.

Evidentemente, per fare un problema più convincente occorre estendere l'elaborazione vista per un impiegato ad n impiegati.

Esprimiamo con l'indice "i" la successione degli stipendi considerati. Logico che i partirà col valore uno, aumenterà di 1 ogni volta che ripetiamo l'operazione fino a raggiungere il valore n, dopo il quale l'elaborazione s'arresterà sullo stop.

Di questa operazione diamo quattro versioni riportate nella figura 3.5. Come si può vedere le quattro soluzioni sono perfettamente equivalenti.

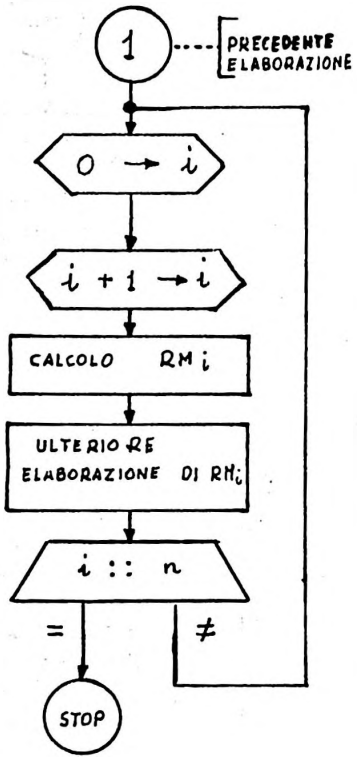
3.9 I cicli iterativi

La parte contrassegnata con (g) è detta "ciclo iterativo". Condizioni necessarie e sufficienti, affinchè si possa inserire un ciclo iterativo in un diagramma di flusso sono:

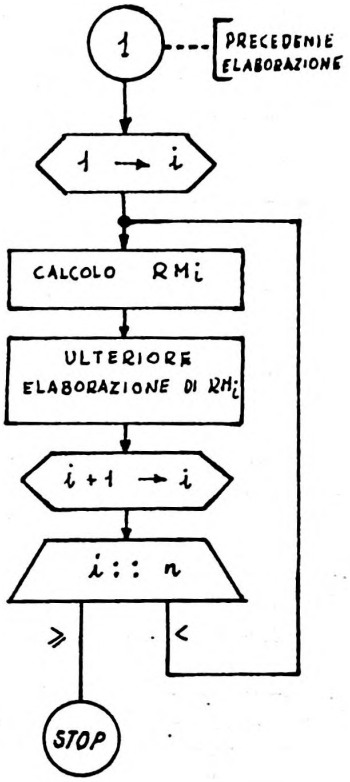
1) La possibilità logica di poter ripetere le medesime operazioni,

FIGURA 3-5

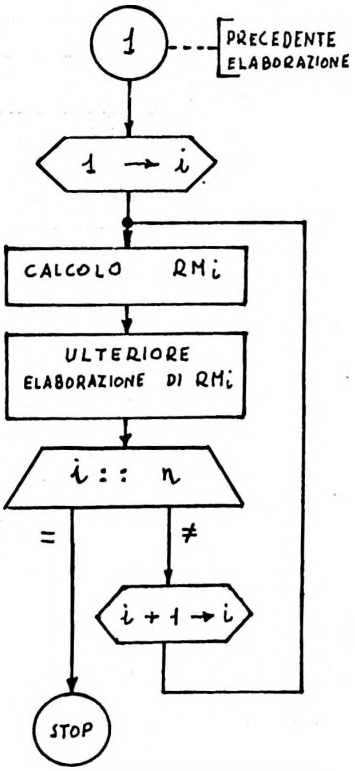
VERSIONE 1



VERSIONE 2



VERSIONE 3



VERSIONE 4

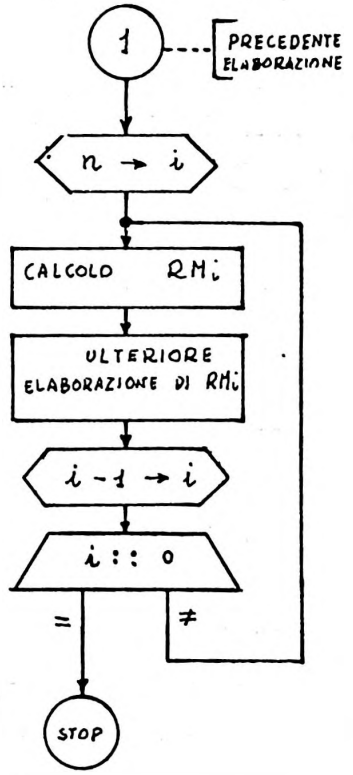
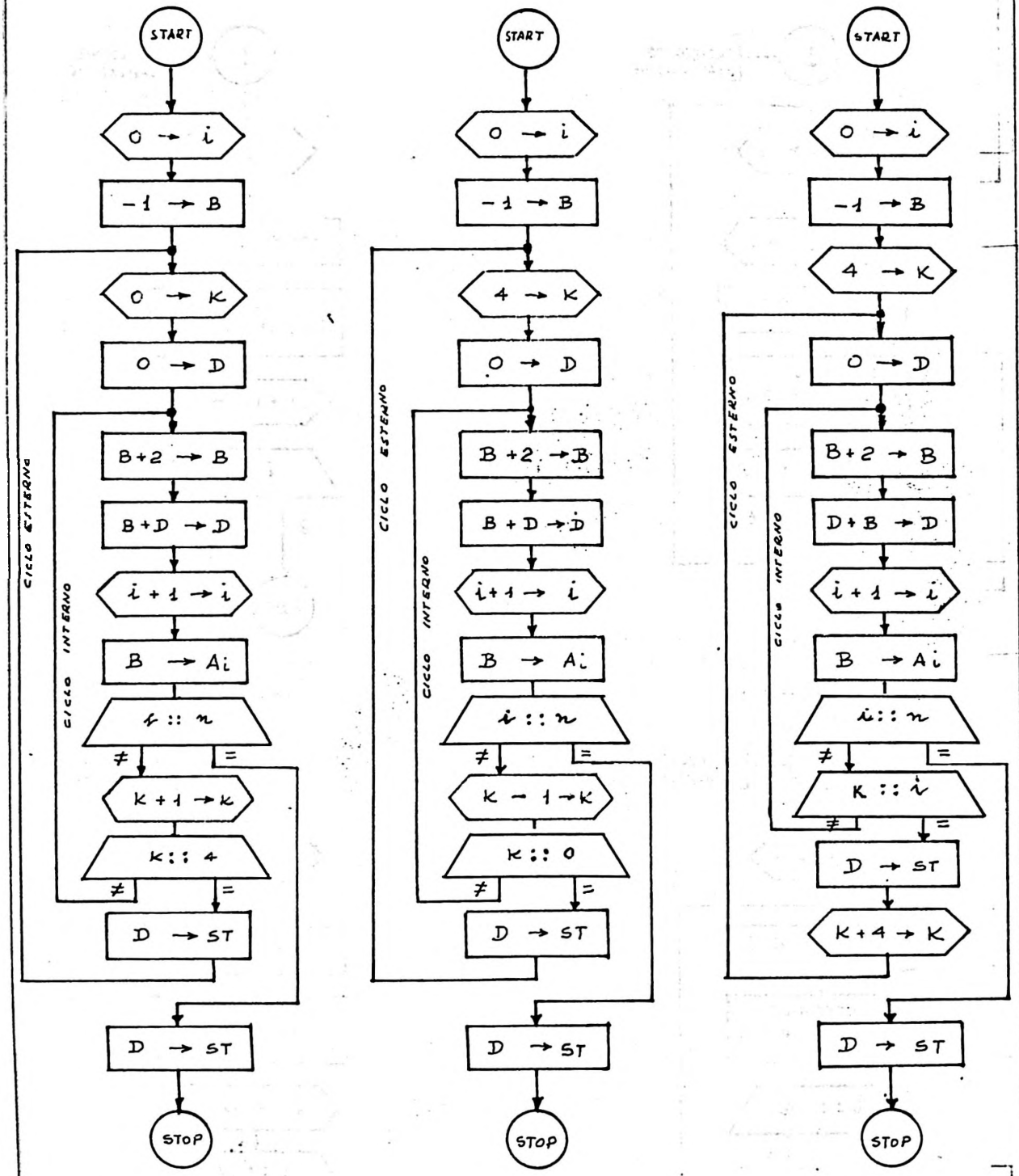


FIGURA 3-6 III

ESEMPI DI DIAGRAMMI RISOLUTIVI



LEGENDA :

Ai = TERMINE GENERICO DELLA SERIE

B = COSTRUZIONE NUMERI DISPARI ; K = CONTATORE CICLO INTERNO

D = SOMMATORE DELLE "QUARTINE" ; ST = STAMPA

2) l'esistenza di una condizione d'uscita.

Si può infatti osservare che senza la condizione d'uscita il ragionamento circolerebbe all'infinito e anche la macchina a sua volta proseguirebbe nell'esecuzione senza mai fermarsi.

I cicli iterativi possono essere di diverso ordine.

E' infatti assai raro un programma in cui compaia un solo ciclo iterativo; mentre è più frequente il caso in cui i cicli sono più di uno e il primo risulta compreso in un secondo più ampio, questo secondo in un terzo e così via.

Supponiamo di risolvere il problema seguente.

Creare una serie di n numeri dispari consecutivi e inviare alla stampa le rispettive somme di 4 in 4. Cioè a dire la somma dei primi 4, poi la somma dei secondi 4 ecc. la serie creata è per n = 8

1, 3, 5, 7, 9, 11, 13, 15.

alla stampa dovrà andare

16; 48 .

La soluzione deve valere anche per n non multiplo di 4, allora alla stampa andrà la somma dell'ultima "quartina" incompleta. Per esempio con n = 10 sia questa la serie creata:

1, 3, 5, 7; 9, 11, 13, 15; 17, 19.

in tal caso andranno alla stampa: 16 ; 48 ; 36.

La figura 3.6 mostra tre diverse soluzioni del problema.

Dagli esempi visti pocanzi si può notare la tendenza ad esprimere le operazioni in forma simbolica e non descrittiva, questo è un fatto puramente opzionale.

Il programmatore è libero di usare l'una o l'altra forma a suo

piacere; si fa rilevare però che l'uso della forma simbolica è bene che sia preceduto da una legenda o glossario, che chiarisca il significato dei simboli impiegati.

3.10 Prova logica dei diagrammi di flusso

Impostato il diagramma di flusso, è necessario procedere a verificarlo mediante una prova logica per vedere, se la linea operativa tracciata è esatta oppure contiene degli errori.

Questa necessità è tanto più evidente, quanto più lungo e laborioso si presenta il flow-chart.

L'importanza della prova logica è fondamentale, sarebbe grave imprudenza passare un diagramma di flusso alla codifica senza prima averlo verificato a fondo. Il programmatore deve ricordare che solo la prova logica potrà dargli la sicurezza della procedura risolutiva; essa quindi non deve essere omessa per nessuna ragione

La prova logica consiste nell'ipotizzare dei dati campione, calcolare in essi le operazioni del problema per via separata, quindi seguire "passo passo" le funzioni espresse dal diagramma di flusso sino ad arrivare allo stop; alla fine il confronto fra i risultati precedentemente calcolati e quelli a cui si è pervenuti col flow-chart stabilirà se il tracciato è esatto.

Quando il procedimento di prova è fatto attentamente, può suggerire correzioni da apportare. La prova logica si suddivide pertanto nelle quattro fasi seguenti:

1) Stesura dei dati campione

   Essa deve essere fatta tenendo presente tutte le particolarità (anche quelle marginali), che i dati del problema possono presentare. Nell'esempio precedente occorrerà ipotizzare due serie campione; una multipla di 4 e una seconda non multipla di 4. Ovviamente non è necessario fare una serie lunghissima: per esempio n = 8 termini nel primo caso, 10 nel secondo sono perfettamente sufficienti.
   Ciò dimostra che i dati campione non devono necessariamente essere numerosi, purchè siano significativi.

2) Calcolo dei risultati per altra via

Tenendo presente l'enunciato del problema, si vedrà di calcolare i risultati per via diversa da quella del diagramma di flusso. Nel senso che, pur svolgendo le stesse operazioni, si fa come se esso non esistesse affatto. Nel caso del problema precedente i risultati si calcolano mentalmente e si mettono in evidenza scrivendoli in un appunto a parte. (Se il problema fosse più complesso il calcolo potrà essere effettuato con l'ausilio di calcolatrici, prospetti ecc.).

3) Prova logica effettiva

Si parte dalla posizione d'inizio (start) e si elabora, sui dati ipotizzati, eseguendo una a una le operazioni indicate dal diagramma, questo fintantochè non si perviene allo stop.

Occorre eseguire le operazioni con calma e con attenzione, al fine di non commettere degli errori di prova, che potrebbero senz'altro venire confusi per errori logici di flow-chart.

4) Controllo dei risultati

Consiste nel verificare la corrispondenza fra i risultati della prova e quelli ottenuti, calcolandoli per via diversa da quella del diagramma. In caso di uguaglianza avremo la certezza dell'esatta impostazione logica del problema e si potrà dire, che il "flow gira".

3.11 Esempio di prova logica

Supponiamo di aver risolto questo problema.

Data una serie di n numeri ottenere le successive somme di 3 in 3. Ciascun totale sarà il termine di una nuova serie, distinta dalla prima n non è necessariamente multiplo di 3, perciò l'ultimo termine non sarà sempre il totale di 3 numeri. Il diagramma di flusso è riportato nella figura 3.7.

Eseguiamo la prova logica.

1) Cominciamo coll'ipotizzare i dati campione.

Evidentemente ci vorranno due serie campione; una con n multiplo, un'altra con n non multiplo di tre. Considerarne una sola sarebbe grave errore, in quanto potrebbe darsi che il flow-chart, pur risultando in quel caso esatto, non lo fosse in un'ipotesi diversa prevista dal problema.

Primo caso: serie di 6 termini (non è necessario fare una campionatura troppo lunga).

4, 6, 1; 3, 0, 9.

Secondo caso: n = 8, la serie sarà per esempio:

5, 2, 1; 8, 7, 9; 11, 3.

2) Calcoliamo, senza prendere in considerazione il diagramma di flusso, i risultati a cui il problema vuole che si pervenga. Essi consistono in una nuova serie di numeri del tipo:

11, 12; per il primo caso e

8, 24; 14 per il secondo caso.

3) Consideriamo i dati di partenza della prima ipotesi ed eseguiamo una per una le operazioni segnate dal flow-chart, seguendo la direzione delle freece fino ad arrivare allo stop.

Il procedimento deve essere fatto in questo modo.

Si mettono in evidenza i dati con tutte le variabili e i parametri del problema, che figurano nel diagramma di flusso, per esempio mediante uno schema del tipo seguente :

n 6

Ai 4, 6, 1; 3, 0, 9

i

K

D

Bj

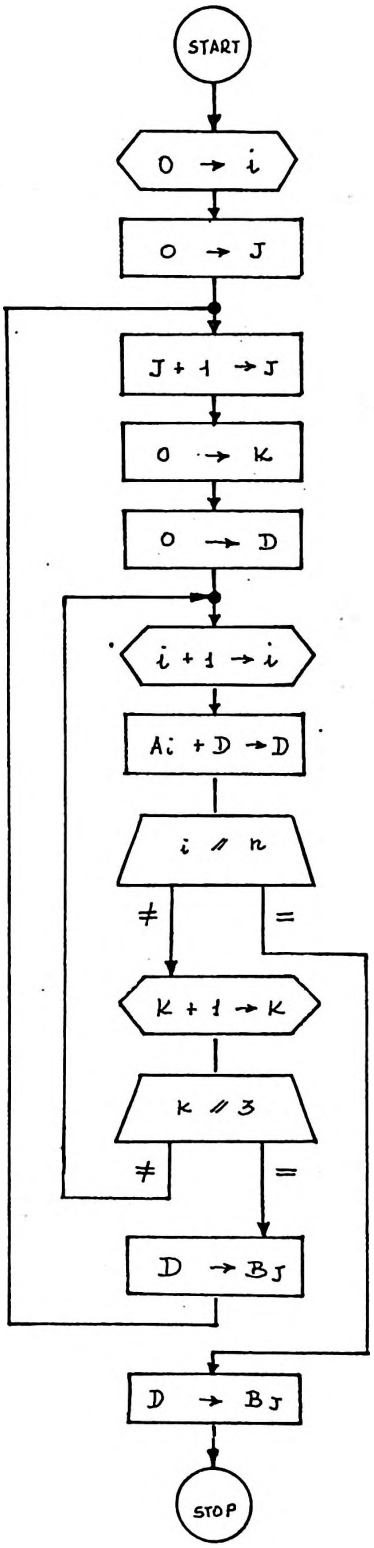
J

FIGURA 3-7 III

LEGENDA

A_i = TERMINE GENERICO DELLA SERIE DATA ; K = CONTATORE DELLE SOMME

B_J = TERMINE GENERICO DELLA SERIE RISULTATO ; D = TOTALIZZATORE SOMME



Si parte dallo start e si cominciano a fare le operazioni:

$0 \longrightarrow i;$

$0 \longrightarrow j;$

$J+1 \longrightarrow J;$

$0 \longrightarrow K;$

$0 \longrightarrow D.$

I nuovi valori e valori successivi assunti dai paramentri del diagramma vengono di volta in volta indicati nel prospetto, in questo modo:

n	6
Ai	4, 6, 1; 3, 0, 9 .
i	0
K	0
D	0
Bj	
J	Ø 1

Quindi si eseguono le operazioni:

$i + 1 \longrightarrow i$

$Ai + D \longrightarrow D,$

in base alle quali i da 0 passa al valore 1; e D da 0 passa a 4, infatti gli si somma Ai, cioè A_1, che è appunto 4.

E' opportuno osservare che l'operazione effettuata a sinistra della freccia trasmette il proprio risultato alla variabile indicata a destra.

Il confronto di i con n, sarà il confronto del valore che ha in quel momento i (cioè 1) con il valore 6 di n. Il risultato è "diverso", faremo pertanto la via $K + 1 \rightarrow K$ che passa ad

1. K successivamente confrontato con 3 ci darà risultato "diverso", per ciò si salirà a ripetere i + 1 → i, ed i assumerà valore 2.

Il prospetto viene aggiornato, operazione per operazione, del nuovo valore delle sue variabili e dei suoi parametri; così a questo punto esso si presenterà:

n	6
Ai	4, 6, 1; 3, 0, 9
i	Ø 1 2
K	Ø 1
D	Ø 4
Bj	
J	Ø 1

Procederemo di questo passo fino ad arrivare all'operazione di STOP. Allora la nostra prova logica per la prima serie di dati campione sarà terminata. Impiegheremo la prova per la seconda serie campione ipotizzata, finita la quale passeremo al confronto dei risultati.

4) Controllo dei risultati. Avviene confrontando dei valori di Bj ricavati dalla prova coi risultati elaborati in sede 2). Se entrambi saranno stati uguali rispettivamente a 11, 12 nel primo caso e a 8, 24, 14 nel secondo, il diagramma di flusso sarà esatto. Diversamente il programmatore dovrà modificarlo opportunamente in modo da correggere gli errori del tracciato, dopo di che tornerà a fare di nuovo la prova logica.

Ripetiamo che quando la prova è eseguita scrupolosamente è di valido aiuto nel suggerire le correzioni da apportare.

3.12 Ottimizzazione di un diagramma di flusso

Il concetto di ottimizzazione di un diagramma di flusso è abbastanza complesso, in quanto si deve tener conto di circostanze particolari al problema in esame, quindi in base a quelle procedere all'ottimizzazione più opportuna.

Ottimizzare significa infatti :

1) Ridurre i tempi logici di esecuzione della linea risolutiva.

2) Ridurre il numero delle operazioni.

3) Rendere il programma più parametrico, cioè idoneo a funzionare per situazioni via via mutevoli.

Quando sia possibile obbedire senza contrasti a tutte e tre queste caratteristiche, non si presenta nessun problema di scelta.

Molto spesso invece può accadere, che, ad esempio, per guadagnare tempo, si deve aumentare il numero totale delle operazioni .

Del problema in figura 3.6 abbiamo tracciato ben tre vie risolutive. Si può osservare che, mentre la prima e la seconda sono equivalenti, la terza invece risulta migliore in senso assoluto. Perciò mentre il numero di operazioni è lo stesso, il tempo di esecuzione è più ridotto, perchè abbiamo tolto al ciclo interno l'operazione K + 1→ K e aggiunto al ciclo esterno l'operazione K + 4→ K. Infatti si è sostituita una operazione che veniva ripetuta n volte, con un'operazione che viene eseguita solo n/4 volte!

In linea di massima si può dire che il programma migliore è quello più veloce. Ma questo non è sempre vero.

Può capitare che il punto 1 ceda al punto 2, allorquando il programma da eseguire contenga un numero elevato di operazioni che stentano a stare tutte quante in memoria.

Infatti in questa ipotesi, se il programma ha una periodicità nel tempo relativamente bassa, può essere conveniente ridurre

il numero delle operazioni, appesantendo i tempi di esecuzione.

Il punto 3 invece è abbastanza delicato. Infatti implica una visione dinamica dei problemi che si identifica colla capacità del programmatore di prevedere nel programma alcune situazioni mutevoli e rendere la linea operativa idonea a tenerne conto.

Facciamo un esempio. Supponiamo che il programmatore debba risolvere un problema di provvigioni ai rappresentanti di una ditta commerciale in grande espansione. La procedura odierna presenta determinate caratteristiche, per esempio delle percentuali scalari di questo tipo: prezzo pieno, provvigione 7%,
sconto 3% " 6%,
sconto 5% " 5%,
ecc.

Potrebbe capitare che in un futuro non molto lontano questi elementi di base venissero a mutare per circostanze varie di mercato, un programma strutturato in modo da prevedere questi cambiamenti, magari appesantendo leggermente la linea operativa, risulterebbe senz'altro ottimizzato.

Il programmatore valuterà queste circostanze, concordando con l'analista, quelle modifiche che si rendessero necessarie nello organigramma di base e nel tracciato informazioni per conseguire la realizzazione di soluzioni più "parametriche" e più "elastiche".

3.13 Differenti livelli di diagrammi di flusso

I diagrammi di flusso possono presentarsi ad un livello generale di grandi linee e ad un livello particolare, dove le operazioni sono trattate elementarmente. Nel primo caso vengono sintetizzate le operazioni fondamentali e trascurati i dettagli ; nel secondo i dettagli sono messi in particolare evidenza.

Possiamo dire che il diagramma a grandi linee tende ad evidenziare il "grosso del problema", dando risalto alle operazioni di maggiore importanza. Questo è visibile nella figura 3.2 nella quale sono rappresentati entrambi i livelli di diagramma, con

riferimento al problema: come mi alzo al mattino.

Dire quale dei due livelli sia preferibile tracciare significherebbe porre una scelta al programmatore, che non sempre è bene che esista. Infatti se si può obiettare che un "flow" generale è troppo sintetico per le operazioni di codifica, tuttavia esso ha il pregio di sottolineare gli aspetti fondamentali di un problema, che un "flow" particolareggiato può finire per perder di vista.

A nostro avviso dato che entrambi presentano pregi e difetti, è preferibile, specialmente per grosse fasi di lavoro, stendere prima un diagramma a grandi linee, in modo da fissare bene i punti vitali del problema; quindi, dopo averlo verificato (la verifica è abbastanza semplice e immediata) tracciare un diagramma più particolareggiato, che tenga conto di operazioni più semplici e più elementari, e perciò sia più rigoroso.

E S E R C I Z I

1) Perchè è necessario tracciare una linea operativa logica?

2) Su che cosa ci si deve basare nella stesura di un diagramma di flusso?

3) Quali simboli sono applicati nella compilazione di un diagramma di flusso?

4) Cos'è una separazione?

5) Quali tipi di separazione conoscete?

6) Eseguite i flow-chart di questi problemi con la prova logica :

   a) Creare una serie di n numeri dispari consecutivi a partire da 1.

   b) Data una serie di numeri ordinarla da minore a maggiore.

   c) Divisione in virgola fissa per sottrazioni successive .

   d) Premesso che ciascun quadrato è date dalla somma di tanti numeri dispari quanto è il valore della base, creare la serie dei primi n quadrati.

7) Come si esegue la prova logica di un diagramma?

8) Cosa significa ottimizzare un flow-chart?

9) Cos'è un flow-chart parametrico?

10) Perchè abbiamo due differenti livelli di diagrammi?

IV I SOTTOPROGRAMMI O SUBROUTINES

4.1. Introduzione e definizione

Si è detto che la logica di macchina è strutturata in modo da eseguire singolarmente operazioni semplici ed elementari.

Se si vogliono eseguire operazioni complesse come ad esempio il calcolo di una divisione in virgola mobile, del seno di un angolo, di una potenza, di un determinante ecc. è evidente che si dovranno combinare sequenzialmente e ciclicamente gruppi di operazioni semplici idonee a svolgere tali calcoli.

In sostanza non esiste un comando singolo e unitario idoneo a svolgere da solo operazioni del genere. Tutti i calcolatori la eseguono, combinando diverse istruzioni di macchina.

Supponiamo per esempio di dover eseguire un programma che in cinque punti diversi prevede il calcolo di radice quadrata.

La prima figura 4.1. dimostra come i punti a,b,c,d,e, costituiscono in sè dei piccoli programmi, ognuno dei quali risolve lo stesso lavoro: il calcolo della radice quadrata di un numero.

Questa specie di "programma nel programma" può essere opportunamente isolato in modo da evitare di inserire ben cinque volte la sequenza di istruzioni di calcolo, risparmiando in questo modo spazio di programma (seconda figura 4.1.).

E' chiaro che nel momento in cui il Programma principale ha necessità di eseguire il calcolo di radice quadrata non farà che saltare a quella "routine", la quale a sua volta dovrà rientrare al programma principale al punto in cui si è saltati.

La "routine" di istruzioni così isolata è detta "subroutines" o "sotto programma".

Definiamo dunque sottoprogramma una sequenza in grado di svolgere un lavoro proprio e determinato, collegata col programma principale in modo che questo possa accedervi da punti diversi, ai quali il sottoprogramma è in grado di ricollegarsi di volta in

volta, dopo aver eseguito il lavoro.

L'uso dei sottoprogrammi presenta la necessità di ricordare il punto in cui si è lasciato il programma principale in modo da poter rientrare.

Ciò implica alcuni tecnicismi di collegamento, che giustificano l'isolamento del sottoprogramma solo se il programma principale deve richiamarlo da più punti diversi.

4.2 Caratteristiche dei sotto programmi

Un sottoprogramma può essere "statico" oppure "dinamico".

E' detto statico quando viene usato così com'è senza alcun adattamento alle esigenze particolari del problema. E' invece "dinamico", se la sua messa a punto implica dei parametri supplementari, che devono essergli comunicati prima di chiamarlo in funzione, in quanto essi possono modificarne lo svolgimento.

Es. Il calcolo della divisione in virgola fissa dà luogo a un sottoprogramma statico. Il calcolo della divisione in virgola mobile è per contro un sottoprogramma dinamico, in quanto occorre volta per volta precisargli il numero di cifre decimali desiderate.

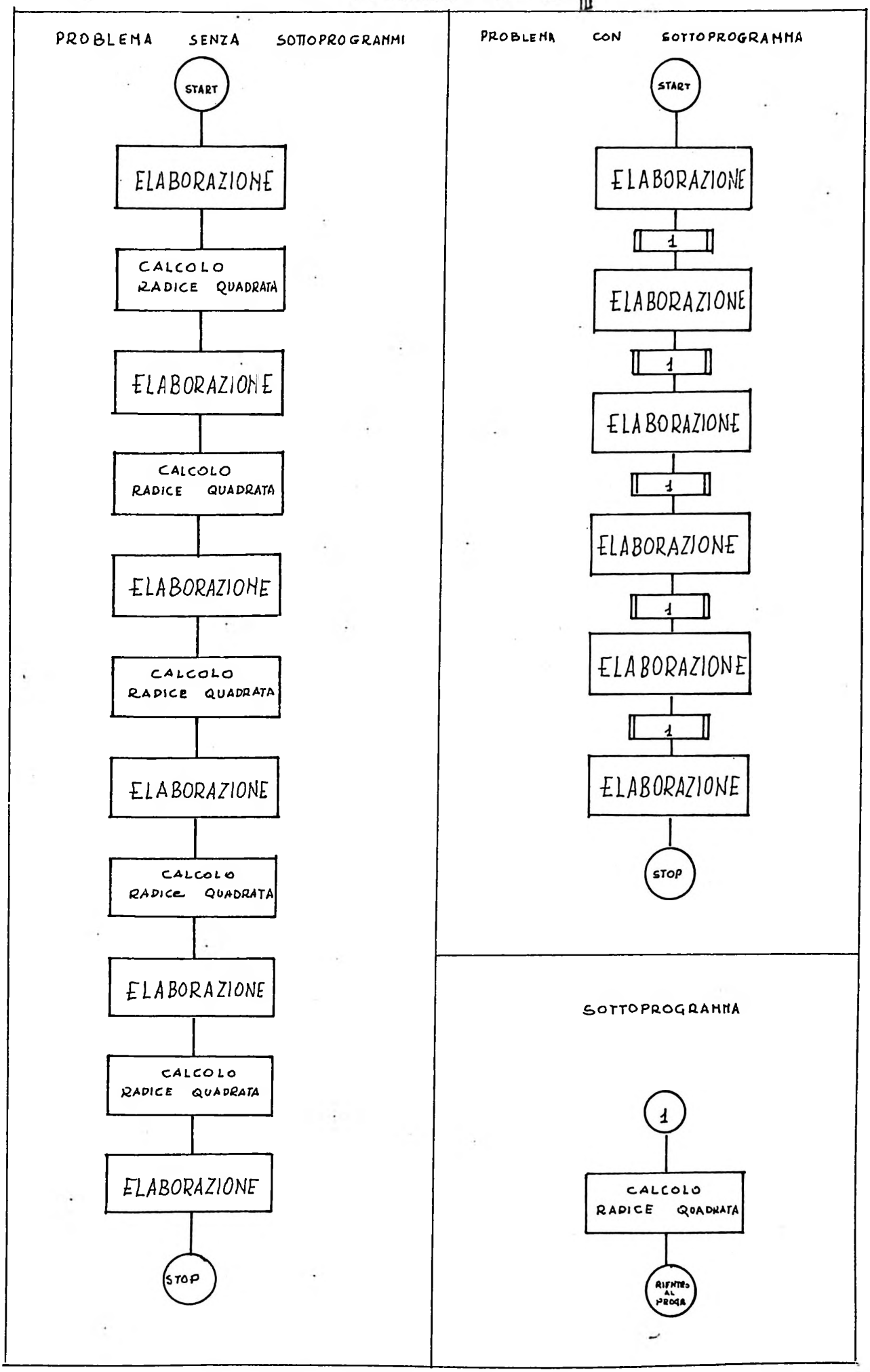
4.3 Biblioteche di sotto programmi

Vi è un gran numero di problemi scientifici e logici che hanno un carattere sufficientemente generale per meritare di essere programmati una volta sola.

L'insieme di tutti questi problemi riuniti assieme costituisce una biblioteca di sottoprogrammi, della quale il programmatore si serve direttamente in tutti i casi in cui gli si presenta la necessità.

La raccolta di questi argomenti in biblioteche di sottoprogrammi è particolarmente utile per il "centro di elaborazione", consentendo di eliminare molte ripetizioni di lavori dello stesso tipo e di dare omogeneità a tutte le procedure.

FIGURA 4-1


PROBLEMA SENZA SOTTOPROGRAMMI
START
ELABORAZIONE
CALCOLO RADICE QUADRATA
ELABORAZIONE
CALCOLO RADICE QUADRATA
ELABORAZIONE
CALCOLO RADICE QUADRATA
ELABORAZIONE
CALCOLO RADICE QUADRATA
ELABORAZIONE
CALCOLO RADICE QUADRATA
ELABORAZIONE
STOP
PROBLEMA CON SOTTOPROGRAMMA
START
ELABORAZIONE
1
ELABORAZIONE
1
ELABORAZIONE
1
ELABORAZIONE
1
ELABORAZIONE
1
ELABORAZIONE
STOP
SOTTOPROGRAMMA
1
CALCOLO RADICE QUADRATA
RIENTRO AL PROGR

E S E R C I Z I

1) Cos'è un sottoprogramma?

2) Quali sono le condizioni logiche che si richiedono, perchè si possa fare una subroutine?

3) Cos'è un sottoprogramma dinamico?

4) Una routine di sottoprogramma fa risparmiare tempo di esecuzione?

5) Quale tecnicismo di collegamento richiede un sottoprogramma?

6) Utilità di una biblioteca di sottoprogrammi?

7) Eseguite il flow chart, dove sei numeri A,B,C,D,E,F, devono essere divisi per altrettanti differenti numeri pari consecutivi a partire da 2, i quozienti o i resti formeranno due distinte serie di 6 numeri ciascuna. L'operazione di divisione deve essere calcolata con sottoprogramma.

V LINEA OPERATIVA DI MACCHINA E CODIFICAZIONE

5.1 Introduzione

Una volta preparato e provato il diagramma a blocchi, occorre procedere alla stesura della linea operativa di macchina, cioè codificare il programma. La sequenza di operazioni descritte dal flow-chart, viene tradotta in una sequenza di istruzioni di macchina, il cui insieme costituirà il programma vero e definitivo.

Risulta subito chiaro che la procedura di codifica presuppone la conoscenza specifica dell'Elaboratore usato.

I linguaggi di codifica

La codifica può essere scritta in due differenti tipi di linguaggio:

1) linguaggio base, detto anche linguaggio di macchina,

2) linguaggi simbolici.

5.2 Il linguaggio base

Scrivere un programma in linguaggio base significa usare gli stessi codici (caratteri) intelleggibili alla macchina al momento in cui dovrà eseguirlo.

Per questo la programmazione in linguaggio base non può evidentemente prescindere dalle conoscenze particolari inerenti al tipo di Elaboratore in oggetto.

Questo tipo di linguaggio, se presenta il vantaggio di poter direttamente registrare in macchina il programma che si è preparato, tuttavia è fonte di molti inconvenienti:

1) Richiede una conoscenza molto profonda dell'Elaboratore e di tutto il sistema integrato, di cui fa parte; conoscenza che non può prescindere da nozioni di carattere tecnico-funzionale.

Sopratutto si dovrà conoscere molto bene tutte le istruzioni pertinenti a quella macchina.

2) Il procedimento di codifica risulta pesante e laborioso, richiede la continua consultazione di tabelle per la conoscenza di diversi caratteri operativi, delle particolarità delle istruzioni, che possono sfuggire alla memoria dell'individuo e si presenta alquanto rigoroso ed ermetico.

3) Esso toglie al programma quella elasticità e quel dinamismo che si rendono necessari ogni qual volta occorre portare correzioni e modifiche di qualsiasi genere, in sede stessa di compilazione del programma.

Queste ragioni hanno favorito la ricerca di forme nuove intese ad eliminare tutti gli inconvenienti, che per un Centro di programmazione si traducono ovviamente in un maggior costo di addestramento e di aggiornamento del personale e in una grossa perdita di tempo durante la compilazione del programma, a tutto svantaggio della economia del Centro stesso.

5.3 I linguaggi simbolici

I linguaggi simbolici hanno la proprietà di esprimere i comandi in maniera assai vicina al linguaggio corrente, per cui la compilazione del programma risulta molto facile per il codificatore, nel senso che vengono a mancare il rigore, l'ermetismo e la rigidità propria del linguaggio di base.

L'adozione di tali linguaggi consente di fare a meno di molte conoscenze particolari di macchina, se non in certi casi, di ignorarle addirittura. Inoltre per i linguaggi a livello elevato la compilazione del programma di macchina risulta assai più breve e più sintetica.

Questi linguaggi però presumono l'esistenza di programmi traduttori, che rendono la macchina capace di intendere i loro simbolismi.

Di qui una doppia classificazione: la prima considera i linguaggi da un punto di vista logico, la seconda dal punto di vista dei programmi traduttori.

5.4 Linguaggi simbolici da un punto di vista logico

Essi si presentano a due differenti livelli, i quali definiscono il grado di programmazione automatica; vogliamo parlare del livello "uno - uno" o "low level" e di livelli più elevati o "high level".

Il primo è chiamato "uno-uno" perchè presenta una corrispondenza biunivoca fra le istruzioni, dette frasi, del linguaggio simbolico e le istruzioni effettive di macchina: un programma del genere ha lo stesso numero di istruzioni che avrebbe,se fosse redatto in linguaggio base.

Quando si passa invece a linguaggi simbolici a livello più elevato, ad una loro "frase" possono corrispondere più istruzioni di base, che normalmente sono un'intera subroutine.

Nel linguaggio "uno-uno", dato lo stretto rapporto tra le frasi e le istruzioni di base, non si può prescindere da un discreto numero di conoscenze tecnico-funzionali dell'Elaboratore in oggetto, specialmente per quanto riguarda la parte istruzioni . Mentre per i linguaggi a livello più elevato queste nozioni possono essere tranquillamente ignorate dal programmatore.

5.5 I linguaggi simbolici dal punto di vista dei programmi traduttori

Essi si suddividono in due gruppi:

1) programmi compilatori

2) programmi interpretatori.

A loro volta tra i programmi compilatori distinguiamo gli assemblatori.

I compilatori. I programmi simbolici sottoposti alla procedura di compilazione vengono prima tradotti in linguaggio base,quindi successivamente vengono eseguiti. Si noti che il momento dell'esecuzione può essere enormemente differito rispetto a quello

della traduzione, ragione per cui questa operazione viene eseguita una volta per tutte.

Quando il linguaggio simbolico si compone delle stesse operazioni del codice reale, linguaggio uno-uno, semplicemente scritte in una forma più chiara e differente, il programma che lo tradurrà in linguaggio di base è detto assemblatore.

Fra compilazione in senso stretto e assemblaggio esiste una differenza fondamentale. Infatti, il linguaggio tradotto da un assemblatore è un linguaggio simbolico di "low level", mentre quello tradotto da un compilatore in senso stretto è un linguaggio "high level".

Esempi di assemblatori sono il Programma PSICO per l'Elea 9003 e l'APS per l'Elea 6001, mentre il PALGO 6001 e la PAC 9003,sono altrettanti esempi di compilatori per programmi a linguaggio elevato.

Gl'Interpretatori. L' "interpretazione" fa capo ad una tecnica completamente differente dalla compilazione.

La macchina riceve un programma scritto in linguaggio simbolico e l'introduce in memoria, come se fosse un programma in codice base. A questo punto la macchina prende in esame ciascuna istruzione simbolica, l'analizza e l'interpreta per mezzo di un programma detto appunto interpretatore e alla fine l'esegue.

Insomma per il programmatore tutto avviene, some se la macchina fosse rimpiazzata da una nuova macchina dotata di proprietà capaci di intendere il linguaggio simbolico.

Per l'Elaboratore il programma simbolico si presenta come una successione di codici strani, che l'interpretatore gli renderà intelleggibili, attraverso una considerevole collezione di sottoprogrammi, idonei a tradurre la simbologia del programma registrato.

Un esempio di interpretatore per i nostri Elaboratori ci è offerto dal programma "Bell".

E S E R C I Z I

1) Cos'è un linguaggio simbolico?

2) A quale livello può presentarsi un linguaggio simbolico?

3) Differenza fra programma interpretatore e compilatore.

4) Cos'è un assemblatore?

5) Quali inconvenienti presenta il linguaggio base nella codifica?

6) Quali assemblatori e compilatori hanno rispettivamente l'Elea 9003 e l'Elea 6001?

7) A vostro avviso è più pratico un linguaggio interpretatore o compilatore?

8) A vostro avviso è preferibile usare un linguaggio compilatore o assemblatore?